



**TUGAS AKHIR - KI1502**

# **RANCANG BANGUN INTEGRASI APLIKASI SISTEM KEHADIRAN MAHASISWA MENGUNAKAN KERANGKA KERJA LARAVEL STUDI KASUS JURUSAN TEKNIK INFORMATIKA ITS**

**Mohammad Shahbana Satriawan**  
**NRP 5113100050**

**Dosen Pembimbing**  
**Sarwosri, S.Kom.,M.T**  
**Dwi Sunaryono, S.Kom.,M.Kom.**

**JURUSAN TEKNIK INFORMATIKA**  
**Fakultas Teknologi Informasi**  
**Institut Teknologi Sepuluh Nopember**  
**Surabaya 2017**

*[Halaman ini sengaja dikosongkan]*



**TUGAS AKHIR - KI1502**

**RANCANG BANGUN INTEGRASI APLIKASI SISTEM  
KEHADIRAN MAHASISWA MENGGUNAKAN  
KERANGKA KERJA LARAVEL STUDI KASUS  
JURUSAN TEKNIK INFORMATIKA ITS**

**Mohammad Shahbana Satriawan  
NRP 5113100050**

**Dosen Pembimbing  
Sarwosri, S.Kom.,M.T  
Dwi Sunaryono, S.Kom.,M.Kom.**

**JURUSAN TEKNIK INFORMATIKA  
Fakultas Teknologi Informasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2017**

*[Halaman ini sengaja dikosongkan]*



**UNDERGRADUATE THESES - KI1502**

# **DESIGN OF APPLICATION INTEGRATION OF STUDENT ATTENDANCE SYSTEM USING LARAVEL FRAMEWORK CASE STUDY OF INFORMATICS DEPARTMENT ITS**

**Mohammad Shahbana Satriawan  
NRP 5113100050**

**Supervisors  
Sarwosri, S.Kom.,M.T  
Dwi Sunaryono, S.Kom.,M.Kom.**

**DEPARTMENT OF INFORMATICS  
Faculty of Information Technology  
Institut Teknologi Sepuluh Nopember  
Surabaya 2017**

*[Halaman ini sengaja dikosongkan]*

## **LEMBAR PENGESAHAN**

**Rancang Bangun Integrasi Aplikasi Sistem Kehadiran  
Mahasiswa Menggunakan Kerangka Kerja Laravel Studi  
Kasus Jurusan Teknik Informatika ITS**

### **TUGAS AKHIR**

Diajukan Untuk Memenuhi Salah Satu Syarat  
Memperoleh Gelar Sarjana Komputer  
pada  
Bidang Studi Rekayasa Perangkat Lunak  
Program Studi S-1 Jurusan Teknik Informatika  
Fakultas Teknologi Informasi  
Institut Teknologi Sepuluh Nopember

Oleh

**MOHAMMAD SHAHBANA SATRIAWAN**  
**NRP: 5113 100 050**

Disetujui oleh Dosen Pembimbing Tugas Akhir

Sarwosri, S.Kom., M.T.  
(NIP 197608092001122001)

Dwi Sunaryono, S.Kom., M.Kom.  
(NIP 197205281997021001)



**SURABAYA**  
**MEI 2017**

*[Halaman ini sengaja dikosongkan]*



# **Rancang Bangun Integrasi Aplikasi Sistem Kehadiran Mahasiswa Menggunakan Kerangka Kerja Laravel Studi Kasus Jurusan Teknik Informatika ITS**

**Nama Mahasiswa : Mohammad Shahbana Satriawan**  
**NRP : 5113 100 050**  
**Jurusan : Teknik Informatika FTIf-ITS**  
**Dosen Pembimbing 1 : Sarwosri, S.Kom.,M.T**  
**Dosen Pembimbing 2 : Dwi Sunaryono, S.Kom.,M.Kom.**

## **Abstrak**

*Sistem kehadiran mahasiswa merupakan proses pencatatan kehadiran mahasiswa di kelas. Di Teknik Informatika ITS, sistem validasi kehadiran mahasiswa dilakukan secara manual. Hal ini menimbulkan celah kelemahan yaitu terjadi kecurangan dalam sistem kehadiran. Dengan menggunakan validasi kehadiran manual, mahasiswa lain bisa memberi tanda tangan kehadiran untuk mahasiswa lain. Oleh karena itu, diperlukan suatu sistem yang dapat digunakan untuk mengatasi kecurangan dalam sistem kehadiran. Dan juga untuk mengatasi mahasiswa yang sering absen dalam kelas, diperlukan suatu sistem yang dapat memonitor hal tersebut.*

*Dalam tugas akhir ini dirancang sebuah sistem kehadiran berbasis web dan mobile. Proses validasi kehadiran mahasiswa dilakukan melalui pencocokan wajah dan tanda tangan. Integrasi data-data pada sistem ini dibuat dengan kerangka kerja Laravel.*

*Hasil pengujian dan implementasi sistem ini adalah mampu memenuhi kebutuhan di Teknik Informatika mengenai sistem informasi kehadiran mahasiswa. Sistem yang dibangun memudahkan proses pencatatan dan perekapan data kehadiran mahasiswa.*

***Kata kunci: Kehadiran, Laravel, Mobile, Sistem Informasi***

# **Design of Application Integration of Student Attendance System Using Laravel Framework Case Study of Informatics Department ITS**

**Student Name : Mohammad Shahbana Satriawan**  
**NRP : 5113 100 192**  
**Major : Informatics Department FTIf – ITS**  
**Advisor I : Sarwosri, S.Kom.,M.T**  
**Advisor II : Dwi Sunaryono, S.Kom.,M.Kom**

## **Abstract**

*Student attendance system is the process of recording student attendance in class. In Informatics Engineering ITS, student attendance validation system is done manually. This leads to a weakness in the presence of fraud in the attendance system. By using manual attendance validation, other students can sign the attendance for other students. Therefore, we need a system that can be used to overcome the fraud in attendance system. And also to overcome the students who are often absent in the class, required a system that can monitor it.*

*In this final project designed a web-based and mobile presence system. The process of student attendance validation is done through matching face and signature. The integration of data on this system is made with the Laravel framework.*

*The results of implementation of this system is able to fulfill the needs in Informatics Engineering on student attendance information system. The built system facilitates the process of recording and reporting of student attendance data.*

**Keywords: Attendance, Laravel, Mobile, Information System**

*[Halaman ini sengaja dikosongkan]*

## KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Alhamdulillahirabbil'alamin, segala puji bagi Allah SWT, yang telah melimpahkan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan tugas akhir yang berjudul **“Rancang Bangun Integrasi Aplikasi Sistem Kehadiran Mahasiswa Menggunakan Kerangka Kerja Laravel Studi Kasus Jurusan Teknik Informatika ITS”**.

Pengerjaan tugas akhir ini merupakan suatu kesempatan yang sangat baik bagi penulis. Dengan pengerjaan tugas akhir ini, penulis bisa belajar lebih banyak untuk memperdalam dan meningkatkan apa yang telah didapatkan penulis selama menempuh perkuliahan di Teknik Informatika ITS.

Selesaiannya tugas akhir ini tidak lepas dari bantuan dan dukungan beberapa pihak, sehingga pada kesempatan ini penulis mengucapkan syukur dan terima kasih kepada:

1. Allah SWT dan Nabi Muhammad SAW.
2. Mama dan Papa, yang selalu mendoakan penulis dan mendukung setiap pilihan yang penulis ambil.
3. Ibu Sarwosri, S.Kom., MT. selaku pembimbing I yang selalu memberikan arahan dan bimbingan kepada penulis untuk menyelesaikan tugas akhir ini.
4. Bapak Dwi Sunaryono, S.Kom., M.Kom. selaku pembimbing II yang selama ini telah membantu dan membimbing penulis selama pengerjaan tugas akhir.
5. Bapak Dr.Eng Darlis Herumurti, S.Kom.,M.Kom. selaku Kepala Jurusan Teknik Informatika ITS, Bapak Dr. Radityo Anggoro, S.Kom.,M.Sc. selaku koordinator TA, dan segenap dosen Teknik Informatika yang telah banyak memberikan ilmu kepada penulis.

6. Azaria Elvinarosa yang selalu menemani dan memberi semangat kepada penulis
7. Teman-teman Administrator Laboratorium RPL yang telah membantu penulis dalam pengerjaan tugas akhir ini.
8. Teman-teman angkatan 2013 yang telah membantu, berbagi ilmu, menjaga kebersamaan, dan memberi motivasi kepada penulis, serta adik-adik angkatan 2014 dan 2015 yang membuat penulis untuk selalu belajar.
9. Teman-teman UKM Bridge ITS yang menjadi tempat penulis untuk melepas penat, mencari hiburan dan berjuang bersama.
10. Serta semua pihak yang telah turut membantu penulis dalam menyelesaikan tugas akhir ini.

Penulis menyadari bahwa tugas akhir ini masih memiliki banyak kekurangan, sehingga dengan kerendahan hati, penulis mengharapkan kritik dan saran dari pembaca untuk perbaikan ke depannya.

Surabaya, Mei 2017

Mohammad Shahbana Satriawan

## DAFTAR ISI

<b>LEMBAR PENGESAHAN .....</b>	<b>v</b>
<b>Abstrak.....</b>	<b>vii</b>
<b>Abstract.....</b>	<b>ix</b>
<b>KATA PENGANTAR.....</b>	<b>xi</b>
<b>DAFTAR ISI.....</b>	<b>xiii</b>
<b>DAFTAR GAMBAR.....</b>	<b>xx</b>
<b>DAFTAR TABEL .....</b>	<b>xxvii</b>
<b>DAFTAR KODE SUMBER .....</b>	<b>xxxi</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1    Latar Belakang .....	1
1.2    Rumusan Masalah.....	3
1.3    Batasan Masalah .....	3
1.4    Tujuan .....	3
1.5    Manfaat .....	4
1.6    Metodologi Pembuatan Tugas Akhir .....	4
1.7    Sistematika Penulisan Laporan Tugas Akhir .....	6
<b>BAB II TINJAUAN PUSTAKA.....</b>	<b>9</b>
2.1    Penelitian Terkait .....	9
2.2    Teknik Informatika ITS .....	10
2.3    Sistem Kehadiran .....	10
2.4    Laravel .....	11
2.5    JSON .....	12
2.6    SQLite .....	13
2.7    MySQL .....	13

## **BAB III ANALISIS DAN PERANCANGAN SISTEM .....15**

3.1	Analisis.....	15
3.1.1	Analisis Permasalahan.....	15
3.1.2	Deskripsi Umum Sistem.....	17
3.1.3	Spesifikasi Kebutuhan Perangkat Lunak.....	20
3.1.3.1	Kebutuhan Fungsional.....	20
3.1.3.2	Kebutuhan Non-Fungsional.....	21
3.1.3.3	Aktor.....	21
3.1.3.4	Kasus Penggunaan.....	21
3.2	Perancangan .....	49
3.2.1	Perancangan Arsitektur Sistem .....	49
3.2.2	Perancangan Basis Data .....	50
3.2.2.1	Tabel Users .....	50
3.2.2.2	Tabel Mahasiswa .....	52
3.2.2.3	Tabel Dosen.....	52
3.2.2.4	Tabel Orangtua .....	53
3.2.2.5	Tabel Wajah.....	54
3.2.2.6	Tabel Signature.....	54
3.2.2.7	Tabel Matakuliah.....	55
3.2.2.8	Tabel Ruang.....	56
3.2.2.9	Tabel Kelas .....	57
3.2.2.10	Tabel Waktu perkuliahan.....	57
3.2.2.11	Tabel Perkuliahan mahasiswa.....	58
3.2.2.12	Tabel Kehadiran .....	59



3.2.2.1	Consteptual Data Modeling (CDM) .....	60
3.2.2.2	Physical Data Modeling (PDM) .....	61
3.2.3	Perancangan Antarmuka Pengguna.....	62
3.2.3.1	Rancangan Halaman Antarmuka Login Pengguna.....	62
3.2.3.2	Rancangan Halaman Antarmuka Kelola Mahasiswa.....	63
3.2.3.3	Rancangan Halaman Antarmuka Melihat Data Perkuliahan.....	65
3.2.3.4	Rancangan Halaman Antarmuka Mengubah Jadwal Perkuliahan.....	67
3.2.3.5	Rancangan Halaman Antarmuka Melihat Data Kehadiran Mahasiswa .....	69
3.2.3.6	Rancangan Halaman Antarmuka Menambah Peserta Kelas .....	71
<b>BAB IV IMPLEMENTASI .....</b>		<b>73</b>
4.1	Lingkungan Implementasi Perangkat Lunak .....	73
4.2	Implementasi Antarmuka Pengguna .....	74
4.2.1	Implementasi Halaman Antarmuka Login .....	74
4.2.2	Implementasi Halaman Antarmuka Kelola Mahasiswa. .....	76
4.2.3	Implementasi Halaman Antarmuka Kelola Kelas .....	77
4.2.4	Implementasi Halaman Antarmuka Melihat Data Perkuliahan .....	78
4.2.5	Implementasi Halaman Antarmuka Mengubah Jadwal Perkuliahan .....	81

4.2.6 Implementasi Halaman Melihat Data Kehadiran Mahasiswa.....	81
4.2.7 Implementasi Halaman Antarmuka Mengubah Status Kehadiran Mahasiswa .....	83
4.2.8 Implementasi Halaman Antarmuka Menambah Peserta Kelas.....	84
4.3 Implementasi Kasus Penggunaan.....	85
4.3.1 Implementasi Kasus Penggunaan Menambah Mahasiswa.....	85
4.3.2 Implementasi Kasus Penggunaan Mengubah Data Mahasiswa.....	86
4.3.3 Implementasi Kasus Penggunaan Menghapus Data Mahasiswa.....	88
4.3.4 Implementasi Menambah Data Kelas.....	89
4.3.5 Implementasi Kegiatan Validasi Kehadiran.....	96
4.3.6 Implementasi Push Notification .....	99
<b>BAB V PENGUJIAN DAN EVALUASI .....</b>	<b>103</b>
5.1 Lingkungan Pengujian .....	103
5.2 Pengujian Fungsionalitas.....	104
5.2.1 Pengujian Fungsionalitas Pada Aktor Karyawan .....	104
5.2.1.1 Pengujian Fungsionalitas Menambah Data Mahasiswa .....	105
5.2.1.2 Pengujian Fungsionalitas Mengubah Data Mahasiswa .....	107
5.2.1.3 Pengujian Fungsionalitas Menghapus Data Mahasiswa .....	108
5.2.1.4 Pengujian Fungsionalitas Menambah Data Dosen .....	109

5.2.1.5	Pengujian Fungsionalitas Mengubah Data	
Dosen	.....	111
5.2.1.6	Pengujian Fungsionalitas Menghapus Data	
Dosen	.....	111
5.2.1.7	Pengujian Fungsionalitas Menambah Data	
Orang Tua.....		112
5.2.1.8	Pengujian Fungsionalitas Mengubah Data Orang	
Tua	.....	114
5.2.1.9	Pengujian Fungsionalitas Menghapus Data	
Orang Tua.....		117
5.2.1.10	Pengujian Fungsionalitas Menambah Data	
Ruangan	.....	118
5.2.1.11	Pengujian Fungsionalitas Mengubah Data	
Ruangan	.....	119
5.2.1.12	Pengujian Fungsionalitas Menghapus Data	
Ruangan	.....	120
5.2.1.13	Pengujian Fungsionalitas Menambah Data	
Matakuliah	.....	121
5.2.1.14	Pengujian Fungsionalitas Mengubah Data	
Matakuliah	.....	123
5.2.1.15	Pengujian Fungsionalitas Menghapus Data	
Matakuliah	.....	124
5.2.1.16	Pengujian Fungsionalitas Menambah Data	
Kelas	.....	125
5.2.1.17	Pengujian Fungsionalitas Mengubah Data	
Kelas	.....	127

5.2.1.18	Pengujian Fungsionalitas Menghapus Data Matakuliah .....	128
5.2.1.19	Pengujian Fungsionalitas Menambah Peserta Kelas .....	129
5.2.1.20	Pengujian Fungsionalitas Melihat Detail Perkuliahan .....	131
5.2.1.21	Pengujian Fungsionalitas Melihat Rekap Kehadiran .....	132
5.2.2	Pengujian Fungsionalitas Pada Aktor Dosen .....	133
5.2.2.1	Pengujian Fungsionalitas Mengaktifkan Perkuliahan .....	133
5.2.2.2	Pengujian Fungsionalitas Mengakhiri Kelas .....	135
5.2.2.3	Pengujian Fungsionalitas Melihat Detail Perkuliahan .....	136
5.2.2.4	Pengujian Fungsionalitas Melihat Rekap Kehadiran Kelas .....	138
5.2.2.5	Pengujian Fungsionalitas Mengubah Status Kehadiran Mahasiswa.....	139
5.2.2.6	Pengujian Fungsionalitas Mengubah Jadwal Perkuliahan .....	141
5.2.3	Pengujian Fungsionalitas Pada Aktor Mahasiswa....	143
5.2.3.1	Pengujian Fungsionalitas Melihat Perkuliahan Aktif Hari Ini .....	143
5.2.3.2	Pengujian Fungsionalitas Melihat List Perkuliahan Hari Ini.....	144

5.2.3.3	Pengujian Fungsionalitas Melihat Riwayat Kehadiran di suatu kelas.....	145
5.2.4	Pengujian Fungsionalitas Pada Aktor Orang Tua ....	146
5.2.4.1	Pengujian Fungsionalitas Melihat Perkuliahan Anak Minggu Ini .....	146
5.2.4.2	Pengujian Fungsionalitas Monitor Kehadiran Anak .....	147
5.2.5	Pengujian Fungsionalitas Notifikasi .....	148
5.2.5.1	Pengujian Fungsionalitas Notifikasi Ketidakhadiran mahasiswa pada Perkuliahan .....	148
5.2.5.2	Pengujian Fungsionalitas Notifikasi Perubahan Jadwal Perkuliahan.....	149
5.3	Pengujian Penerimaan Pengguna .....	151
5.3.1	Pengujian Penerimaan Pengguna Pada Aktor Mahasiswa.....	151
5.3.2	Pengujian Penerimaan Pengguna Pada Aktor Dosen	153
5.3.3	Pengujian Penerimaan Pengguna Pada Aktor Orang Tua .....	154
5.4	Evaluasi Pengujian.....	156
<b>BAB VI KESIMPULAN DAN SARAN.....</b>		<b>159</b>
6.1	Kesimpulan .....	159
6.2	Saran .....	159
<b>DAFTAR PUSTAKA.....</b>		<b>161</b>
<b>LAMPIRAN.....</b>		<b>163</b>
<b>BIODATA PENULIS.....</b>		<b>167</b>

*[Halaman ini sengaja dikosongkan]*

## DAFTAR GAMBAR

Gambar 2.1 Pertukaran Data melalui RESTful API.....	12
Gambar 3.1 Deskripsi Umum Sistem.....	17
Gambar 3.2 Diagram Kasus Penggunaan .....	22
Gambar 3.3 Diagram Aktifitas Melihat Data Mahasiswa .....	25
Gambar 3.4 Diagram Aktivitas Menambah Data Mahasiswa .....	27
Gambar 3.5 Diagram Aktivitas Mengubah Data Mahasiswa .....	29
Gambar 3.6 Diagram Aktivitas Menghapus Data Mahasiswa.....	30
Gambar 3.7 Diagram Aktivitas Menambah Kelas .....	34
Gambar 3.8 Diagram Aktivitas Mengubah Kelas .....	36
Gambar 3.9 Diagram Aktivitas Menghapus Kelas.....	37
Gambar 3.10 Diagram Aktivitas Melihat Perkuliahan .....	38
Gambar 3.11 Diagram Aktivitas Mengubah Jadwal Perkuliahan .....	40
Gambar 3.12 Diagram Aktivitas Melihat Data Kehadiran .....	41
Gambar 3.13 Diagram Aktivitas Mengisi Status Kehadiran .....	43
Gambar 3.14 Diagram Aktivitas Menambah Peserta Kelas .....	45
Gambar 3.15 Diagram Aktivitas Menghapus Peserta Kelas .....	46
Gambar 3.28 Conceptual Data Modeling .....	60
Gambar 3.29 Physical Data Modeling.....	61
Gambar 3.30 Rancangan Halaman Antarmuka Login Pengguna .....	63
Gambar 3.31 Perancangan Antarmuka Kelola Mahasiswa .....	64
Gambar 3.32 Perancangan Antarmuka Tambah Mahasiswa.....	64
Gambar 3.33 Perancangan Antarmuka Ubah Mahasiswa .....	65
Gambar 3.34 Perancangan Antarmuka Melihat Data Perkuliahan (Aktor: Dosen).....	66
Gambar 3.35 Perancangan Antarmuka Melihat Data Perkuliahan (Aktor: Orang Tua).....	66
Gambar 3.36 Perancangan Antarmuka Melihat Data Perkuliahan (Aktor: Mahasiswa dan Orang Tua).....	67
Gambar 3.37 Perancangan Antarmuka Mengubah Jadwal Perkuliahan (Mobile).....	68

Gambar 3.38 Perancangan Antarmuka Mengubah Jadwal Perkuliahan (Web).....	68
Gambar 3.39 Rancangan Halaman Antarmuka Melihat Data Kehadiran Mahasiswa (Web) .....	69
Gambar 3.40 Rancangan Halaman Antarmuka Melihat Data Kehadiran Mahasiswa (Mobile) .....	70
Gambar 3.41 Rancangan Halaman Antarmuka Menambah Peserta Kelas .....	71
Gambar 4.1 Implementasi Halaman Antarmuka Login Aplikasi Web .....	74
Gambar 4.2 Implementasi Halaman Antarmuka Login Aplikasi Mobile .....	75
Gambar 4.3 Implementasi Halaman Antarmuka Kelola Mahasiswa .....	76
Gambar 4.4 Implementasi Halaman Antarmuka Tambah Mahasiswa .....	76
Gambar 4.5 Implementasi Halaman Antarmuka Kelola Kelas .....	77
Gambar 4.6 Implementasi Halaman Antarmuka Menambah Kelas .....	78
Gambar 4.7 Implementasi Halaman Antarmuka Melihat Data Perkuliahan (Web).....	79
Gambar 4.8 Implementasi Halaman Antarmuka Melihat Data Perkuliahan (Mobile).....	79
Gambar 4.9 Implementasi Mahasiswa Melihat Perkuliahan Hari Ini.....	80
Gambar 4.10 Implementasi Orang Tua Melihat Perkuliahan anak .....	80
Gambar 4.11 Implementasi Halaman Mengubah Jadwal Perkuliahan (Web).....	81
Gambar 4.12 Implementasi Halaman Melihat Data Kehadiran Mahasiswa (Web).....	82
Gambar 4.13 Implementasi Halaman Melihat Data Kehadiran Mahasiswa (Mobile).....	82
Gambar 4.14 Implementasi Halaman Ubah Status Kehadiran Mahasiswa (Web).....	83



Gambar 4.15 Implementasi Halaman Ubah Status Kehadiran Mahasiswa (Mobile).....	84
Gambar 4.16 Implementasi Halaman Antarmuka Menambah peserta Kelas .....	84
Gambar 4.17 Pencocokan lokasi .....	96
Gambar 4.18 Mencocokkan lokasi dengan gambar kelas .....	97
Gambar 5.1 Halaman Tambah Mahasiswa.....	106
Gambar 5.2 Mahasiswa Berhasil Ditambahkan .....	106
Gambar 5.3 Halaman Mengubah Data Mahasiswa .....	107
Gambar 5.4 Perubahan Berhasil Dilakukan .....	108
Gambar 5.5 Menghapus Data Mahasiswa .....	109
Gambar 5.6 Berhasil Menghapus Data Mahasiswa.....	109
Gambar 5.7 Halaman Menambah Data Dosen .....	110
Gambar 5.8 Dosen Berhasil Ditambahkan .....	110
Gambar 5.9 Halaman Ubah Dosen.....	116
Gambar 5.10 Berhasil Menyimpan Perubahan Data Dosen .....	116
Gambar 5.11 Menghapus Data Dosen.....	112
Gambar 5.12 Data Dosen Berhasil Dihapus.....	112
Gambar 5.13 Halaman Menambah Data Orang Tua .....	113
Gambar 5.14 Data Orang Tua Berhasil Ditambahkan.....	113
Gambar 5.15 Halaman Ubah Data Orang Tua .....	115
Gambar 5.16 Berhasil Ubah Data Orang Tua .....	115
Gambar 5.17 Menghapus Data Orang Tua.....	117
Gambar 5.18 Data Orang Tua Berhasil Dihapus.....	117
Gambar 5.19 Halaman Menambah Data Ruangan .....	118
Gambar 5.20 Berhasil Menambah Data Ruangan .....	119
Gambar 5.21 Halaman Mengubah Data Ruangan .....	120
Gambar 5.22 Berhasil Mengubah Data Ruangan .....	120
Gambar 5.23 Halaman Menghapus Data Ruangan .....	121
Gambar 5.24 Berhasil Menghapus Data Ruangan .....	121
Gambar 5.25 Halaman Menambah Matakuliah.....	122
Gambar 5.26 Berhasil Menambahkan Matakuliah.....	122
Gambar 5.27 Halaman Mengubah Matakuliah .....	123
Gambar 5.28 Berhasil Mengubah Data Matakuliah .....	124
Gambar 5.29 Halaman Menghapus Data Matakuliah .....	124

Gambar 5.30 Halaman Menambah Kelas .....	126
Gambar 5.31 Berhasil Menambah Kelas .....	126
Gambar 5.32 Halaman Mengubah Data Kelas .....	127
Gambar 5.33 Berhasil Mengubah Data Kelas .....	128
Gambar 5.34 Halaman Menghapus Data Kelas.....	129
Gambar 5.35 Isi File Excel Contohupload.xls.....	130
Gambar 5.36 Halaman Upload File.....	130
Gambar 5.37 Peserta Kelas Berhasil Ditambahkan .....	130
Gambar 5.38 Melihat Detail Perkuliahan .....	131
Gambar 5.39 Halaman Melihat Rekap Kehadiran.....	132
Gambar 5.40 Halaman Pengaktifan Matakuliah (Web) .....	133
Gambar 5.41 Perkuliahan Berhasil Diaktifkan (Web).....	134
Gambar 5.42 Halaman Mengaktifkan Kuliah (Mobile) .....	134
Gambar 5.43 Perkuliahan Berhasil Diaktifkan (Mobile).....	134
Gambar 5.44 Halaman Mengakhiri Kelas .....	135
Gambar 5.45 Berhasil Mengakhiri Kelas (Web) .....	135
Gambar 5.46 Berhasil Mengakhiri Kelas (Mobile) .....	136
Gambar 5.47 Halaman Melihat Detail Perkuliahan (Web) ..	137
Gambar 5.48 Berhasil Menampilkan Detail Perkuliahan (Web) .....	137
Gambar 5.49 Berhasil Menampilkan Detail Perkuliahan (Mobile) .....	138
Gambar 5.50 Berhasil Menampilkan Rekap Kehadiran (Web)	139
Gambar 5.51 Berhasil Menampilkan Rekap Kehadiran (Mobile) .....	139
Gambar 5.52 Halaman Ubah Status Kehadiran Mahasiswa (Web) .....	140
Gambar 5.53 Ubah Status Kehadiran Mahasiswa (Mobile) .	141
Gambar 5.55 Halaman Pengubahan Jadwal Perkuliahan (Mobile) .....	142
Gambar 5.56 Halaman Perkuliahan Aktif .....	143
Gambar 5.57 Menampilkan List Perkuliahan .....	144
Gambar 5.58 Menampilkan Riwayat Kehadiran .....	145
Gambar 5.59 Melihat Perkuliahan Anak Minggu Ini .....	146
Gambar 5.60 Halaman Monitor Anak .....	147

Gambar 5.61 Mengganti Status Kehadiran Mahasiswa .....	148
Gambar 5.62 Notifikasi kepada Orang Tua.....	149
Gambar 5.63 Pengubahan Jadwal Perkuliahan .....	150
Gambar 5.64 Notifikasi Penggantian Jadwal .....	150
Gambar 0.1 Diagram Sekuens UC-021 .....	163
Gambar 0.2 Diagram Sekuens UC-022 .....	164
Gambar 0.3 Diagram Sekuens UC-023 .....	165
Gambar 0.4 Diagram Sekuens UC-024 .....	166

*[Halaman ini sengaja dikosongkan]*

## DAFTAR TABEL

Tabel 3.1 Kebutuhan Fungsional.....	20
Tabel 3.2 Kasus Penggunaan.....	23
Tabel 3.3 Spesifikasi Kasus Penggunaan UC-001 .....	25
Tabel 3.4 Spesifikasi Kasus Penggunaan UC-002 .....	26
Tabel 3.5 Spesifikasi Kasus Penggunaan UC-003 .....	28
Tabel 3.6 Spesifikasi Kasus Penggunaan UC-004 .....	29
Tabel 3.7 Spesifikasi Kasus Penggunaan UC-0018 .....	33
Tabel 3.8 Spesifikasi Kasus Penggunaan UC-019 .....	35
Tabel 3.9 Spesifikasi Kasus Penggunaan UC-021 .....	36
Tabel 3.10 Spesifikasi Kasus Penggunaan UC-021 .....	38
Tabel 3.11 Spesifikasi Kasus Penggunaan UC-0022 .....	39
Tabel 3.12 Spesifikasi Kasus Penggunaan UC-023 .....	41
Tabel 3.13 Spesifikasi Kasus Penggunaan UC-024 .....	42
Tabel 3.14 Spesifikasi Kasus Penggunaan UC-029 .....	44
Tabel 3.15 Spesifikasi Kasus Penggunaan UC-0030 .....	45
Tabel 3.16 Spesifikasi Kasus Penggunaan UC-035 .....	48
Tabel 3.17 Spesifikasi Kasus Penggunaan UC-036 .....	48
Tabel 3.18 Daftar Request Web Service .....	49
Tabel 3.19 Atribut Tabel Users .....	51
Tabel 3.20 Atribut Tabel Mahasiswa .....	52
Tabel 3.21 Atribut Tabel Dosen .....	53
Tabel 3.22 Atribut Tabel Orang Tua .....	53
Tabel 3.23 Atribut Tabel Wajah.....	54
Tabel 3.24 Atribut Tabel Mahasiswa .....	55
Tabel 3.25 Atribut Tabel Matakuliah .....	55
Tabel 3.26 Atribut Tabel Ruangan .....	56
Tabel 3.27 Atribut Tabel kelas .....	57
Tabel 3.28 Atribut tabel Waktuperkuliahan .....	58
Tabel 3.29 Atribut Tabel Perkuliahanmahasiswa.....	58
Tabel 3.30 Atribut Tabel Kehadiran.....	59
Tabel 4.1 Lingkungan Implementasi Perangkat Lunak.....	73
Tabel 4.2 Penjelasan Kode Sumber 4-1 .....	86
Tabel 4.3 Penjelasan Kode Sumber 4-2 .....	88

Tabel 4.4 Penjelasan Kode Sumber 4-2.....	88
Tabel 4.5 Penjelasan Kode Sumber 4.4.....	92
Tabel 4.6 Penjelasan Kode Sumber 4.5.....	93
Tabel 4.7 Penjelasan Kode Sumber 4.7.....	97
Tabel 4.8 Penjelasan Kode Sumber 4.8.....	99
Tabel 4.9 Penjelasan Kode Sumber 4.9.....	101
Tabel 5.1 Lingkungan Pengujian Fungsionalitas Aplikasi Mobile .....	103
Tabel 5.2 Lingkungan Pengujian Fungsionalitas Aplikasi Web.....	104
Tabel 5.3 Skenario Pengujian Fungsionalitas Menambah Data Mahasiswa.....	105
Tabel 5.4 Skenario Pengujian Fungsionalitas Mengubah Data Mahasiswa.....	107
Tabel 5.5 Halaman Pengujian Fungsionalitas Menghapus Data Mahasiswa.....	108
Tabel 5.6 Skenario Pengujian Fungsionalitas Menambah Data Dosen.....	109
Tabel 5.7 Skenario Pengujian Fungsionalitas Mengubah Data Dosen.....	111
Tabel 5.8 Halaman Pengujian Fungsionalitas Menghapus Data Dosen.....	111
Tabel 5.9 Skenario Pengujian Fungsionalitas Menambah Data Orang Tua.....	112
Tabel 5.10 Skenario Pengujian Fungsionalitas Mengubah Data Orang Tua.....	114
Tabel 5.11 Halaman Pengujian Fungsionalitas Menghapus Data Orang Tua.....	117
Tabel 5.12 Skenario Pengujian Fungsionalitas Menambah Data Ruangan.....	118
Tabel 5.13 Skenario Pengujian Fungsionalitas Mengubah Data Ruangan.....	119
Tabel 5.14 Fungsionalitas Menghapus Data Ruangan .....	120
Tabel 5.15 Skenario Pengujian Fungsionalitas Menambah Data Matakuliah.....	121

Tabel 5.16 Skenario Pengujian Fungsionalitas Mengubah Data Matakuliah.....	123
Tabel 5.17 Halaman Pengujian Fungsionalitas Menghapus Data .....	124
Tabel 5.18 Skenario Pengujian Fungsionalitas Menambah Data Orang Tua.....	125
Tabel 5.19 Skenario Pengujian Fungsionalitas Mengubah Data Orang Tua.....	127
Tabel 5.20 Halaman Pengujian Fungsionalitas Menghapus Data .....	128
Tabel 5.21 Halaman Pengujian Fungsionalitas Menghapus Data .....	129
Tabel 5.22 Halaman Pengujian Fungsionalitas Menghapus Data .....	131
Tabel 5.23 Halaman Pengujian Fungsionalitas Melihat Rekap Kehadiran .....	132
Tabel 5.24 Skenario Pengujian Fungsionalitas Mengaktifkan Perkuliahan.....	133
Tabel 5.25 Skenario Pengujian Fungsionalitas Mengakhiri Kelas .....	135
Tabel 5.26 Skenario Pengujian Fungsionalitas Melihat Detail Perkuliahan.....	136
Tabel 5.27 Skenario Pengujian Fungsionalitas Melihat Rekap Kehadiran Kelas .....	138
Tabel 5.28 Skenario Pengujian Fungsionalitas Mengubah Status Kehadiran Mahasiswa .....	139
Tabel 5.29 Skenario Pengujian Fungsionalitas Mengubah Jadwal Perkuliahan.....	141
Tabel 5.30 Melihat Perkuliahan Aktif Hari Ini .....	143
Tabel 5.31 Melihat List Perkuliahan Hari Ini.....	144
Tabel 5.32 Melihat Riwayat Kehadiran di suatu kelas .....	145
Tabel 5.33 Melihat Perkuliahan Anak Minggu Ini.....	146
Tabel 5.34 Monitor Kehadiran Anak .....	147

*[Halaman ini sengaja dikosongkan]*

*[Halaman ini sengaja dikosongkan]*



## DAFTAR KODE SUMBER

Kode Sumber 4.1 Menambahkan Data Mahasiswa.....	86
Kode Sumber 4.2 Mengubah Data Mahasiswa .....	87
Kode Sumber 4.3 Menghapus Data Mahasiswa.....	88
Kode Sumber 4.4 Javascript untuk mengatur agar tidak ada kode, jadwal dan ruangan yang bertabrakan .....	91
Kode Sumber 4.5 Fungsi pada KelasController .....	93
Kode Sumber 4.6 Fungsi tambah_kelas .....	95
Kode Sumber 4.7 OnClickListener Tombol Pencocokan Lokasi .....	97
Kode Sumber 4.8.....	99
Kode Sumber 4.9 Pengiriman Notifikasi ke Aplikasi Mobile	101
[Halaman ini sengaja dikosongkan]	

*[Halaman ini sengaja dikosongkan]*

## **BAB I**

### **PENDAHULUAN**

Bab ini menjelaskan garis besar tugas akhir yang meliputi latar belakang, tujuan, rumusan, batasan permasalahan, metodologi pembuatan tugas akhir, dan sistematika penulisan.

#### **1.1 Latar Belakang**

Sistem kehadiran mahasiswa merupakan proses pencatatan kehadiran mahasiswa di kelas. Dengan adanya sistem kehadiran maka dapat membuktikan bahwa mahasiswa tersebut telah berada di dalam kelas. Validasi kehadiran dapat dilakukan dengan berbagai cara. Di Teknik Informatika ITS, sistem validasi kehadiran mahasiswa dilakukan secara manual yaitu melakukan tanda tangan di kertas. Hal ini menimbulkan celah kelemahan yaitu terjadi kecurangan [1] dalam sistem kehadiran atau disebut titip absen. Titip absen merupakan kejadian dimana mahasiswa telah absen tanda tangan namun kenyataannya mahasiswa tersebut tidak hadir di kelas. Dengan menggunakan validasi kehadiran manual, mahasiswa lain bisa memberi tanda tangan kehadiran untuk mahasiswa lain. Oleh karena itu, diperlukan suatu sistem yang dapat digunakan untuk mengatasi kecurangan dalam sistem kehadiran.

Perkembangan teknologi informasi di dunia semakin cepat, khususnya pada teknologi perangkat bergerak dan internet. Akses dan pengambilan informasi dikatakan semakin mudah dan cepat diakses melalui penggunaan perangkat bergerak dan web. Perkembangan ini juga mempengaruhi data sebagai validasi menggantikan kode sandi (password). Validasi merupakan sebuah proses yang wajib ada dimana diperlukan sebuah kebenaran data/informasi, salah satunya adalah proses validasi kehadiran. Proses validasi sudah banyak diterapkan seiring

dengan perkembangan teknologi informasi misalnya validasi kehadiran dengan menggunakan alat yang menerima inputan metode *fingerprint*. Namun kelemahan dari penerapan yang sudah ada adalah kurangnya fleksibilitas dari penerapan ini. Fleksibilitas yang dimaksud adalah teknologi yang digunakan tidak bisa berpindah tempat dan diakses kapanpun. Oleh karena itu, tugas akhir ini akan mengimplementasikan sebuah aplikasi mobile berbasis android. Implementasi pada aplikasi mobile diharapkan mampu menambah fleksibilitas dari sistem kehadiran yang sudah ada saat ini.

Aplikasi sistem kehadiran yang akan dibuat akan dibatasi ruang lingkup penggunaannya. Pengguna dari aplikasi ini adalah mahasiswa Teknik Informatika ITS. Proses validasi kehadiran pada aplikasi ini meliputi dua cara, yaitu pengenalan wajah, dan tanda tangan melalui *device* berbasis android. Setelah melakukan salah satu cara validasi kehadiran, akan dilakukan validasi dengan data masing-masing mahasiswa. Dosen dapat memonitor status kehadiran mahasiswa pada kelas tersebut.

Hasil yang diharapkan dari pengerjaan tugas akhir ini adalah berupa aplikasi *mobile* yang menggantikan proses sistem kehadiran mahasiswa dari sistem manual melalui tanda tangan pada kertas menjadi sistem *online* pada aplikasi *mobile*. Selain itu, dengan aplikasi ini diharapkan juga dapat memberikan kebenaran data kehadirann (*data validation*) serta mempermudah pemantauan kehadiran mahasiswa di setiap mata kuliah. Selain itu, aplikasi ini juga dapat memberikan informasi kepada orang tua/wali mahasiswa tentang status kehadiran anaknya. Notifikasi akan dikirimkan secara otomatis ke *device* orang tua/wali jika anaknya tidak masuk dalam perkuliahan. Informasi ini diharapkan dapat membantu orang tua/wali mahasiswa dalam memantau anaknya.

## 1.2 Rumusan Masalah

Rumusan masalah yang diangkat dalam tugas akhir ini dapat dipaparkan sebagai berikut:

1. Bagaimana membangun Aplikasi *Mobile* Sistem Kehadiran Mahasiswa berbasis Android
2. Bagaimana membangun Aplikasi *Web* dan mengintegrasikannya dengan Aplikasi *Mobile* dengan menggunakan RestAPI yang disusun dengan kerangka kerja Laravel
3. Bagaimana membuat notifikasi melalui Firebase Cloud Messaging dari framework Laravel

## 1.3 Batasan Masalah

Permasalahan yang dibahas dalam tugas akhir ini memiliki beberapa batasan, yaitu sebagai berikut:

1. Aplikasi *web* berbasis kerangka kerja Laravel 5.3
2. Aplikasi *mobile* dapat berjalan pada perangkat *mobile* dengan versi minimal Android 4.1 (*Jellybean / API 18*) berkapasitas RAM 2GB.

## 1.4 Tujuan

Tugas akhir ini mempunyai beberapa tujuan, yaitu sebagai berikut:

1. Membangun Sistem Informasi Kehadiran Mahasiswa berbasis Web yang terintegrasi dengan Aplikasi Kehadiran Mahasiswa berbasis Android.
2. Memberikan solusi yang efektif dalam pencatatan, pengawasan dan perekapan kehadiran perkuliahan.

## 1.5 Manfaat

Hasil dari pengerjaan Tugas Akhir ini memiliki manfaat untuk memperbaharui sistem kehadiran mahasiswa dari sistem manual melalui tanda tangan atau *face recogniton* pada kehadiran kelas menjadi sistem. Dengan demikian, diperolehnya data yang benar tentang kehadiran mahasiswa akan lebih jelas dan terperinci sehingga pemantauan oleh dosen dan orang tua terhadap mahasiswanya dan anaknya dalam kehadiran kuliah akan semakin mudah.

## 1.6 Metodologi Pembuatan Tugas Akhir

Tahapan-tahapan yang dilakukan dalam pengerjaan tugas akhir ini adalah sebagai berikut:

1. Penyusunan proposal tugas akhir.

Proposal ini berisi mengenai proposal tugas mengenai deskripsi tugas akhir mengenai rancang bangun integrasi aplikasi Sistem Kehadiran Mahasiswa

2. Studi literatur

Studi literatur yang dilakukan dalam pengerjaan Tugas Akhir ini adalah mengenai implementasi penggunaan kerangka kerja Laravel, Pembuatan API, Integrasi antara Aplikasi *Web* dan Aplikasi *Mobile*. Studi literatur diambil dari buku, internet, maupun materi mata kuliah yang berhubungan dengan metode yang digunakan.

3. Analisis dan desain perangkat lunak

Tahap ini meliputi perumusan kebutuhan fungsional, kebutuhan non-fungsional, kasus penggunaan, diagram aktivitas, diagram kelas, diagram sekuens, rancangan antarmuka pengguna, serta pembuatan rancangan basis data.

#### 4. Implementasi perangkat lunak

Aplikasi ini diimplementasikan dengan menggunakan kakas bantu :

1. Laravel Framework sebagai kerangka kerja utama
2. IDE Android Studio.
3. Database MySQL dan SQL-Yog untuk mengolah
4. Postman, kakas bantu untuk menguji integrasi *web service* dengan sistem yang sudah dibangun.
5. Sublime Text sebagai *text editor*

#### 5. Pengujian dan evaluasi

Pengujian yang dilakukan adalah pengujian *blackbox*, yaitu pengujian untuk mengetahui apakah semua fungsi perangkat lunak telah berjalan sesuai dengan kebutuhan fungsional yang didefinisikan

#### 6. Penyusunan buku tugas akhir

Pada tahap ini dilakukan penyusunan laporan yang menjelaskan dasar teori dan metode yang digunakan dalam tugas akhir ini serta hasil dari implementasi aplikasi perangkat lunak yang telah dibuat. Sistematika penulisan buku tugas akhir secara garis besar antara lain:

1. Pendahuluan
  - a. Latar Belakang
  - b. Rumusan Masalah
  - c. Batasan Masalah
  - d. Tujuan
  - e. Manfaat
  - f. Metodologi Pembuatan Tugas Akhir
  - g. Sistematika Penulisan Laporan Tugas Akhir
2. Tinjauan Pustaka
3. Analisis dan Perancangan Sistem
4. Pengujian dan Evaluasi
5. Kesimpulan dan Saran

## 6. Daftar Pustaka

### 1.7 Sistematika Penulisan Laporan Tugas Akhir

Buku tugas akhir ini bertujuan untuk mendapatkan gambaran dari pengerjaan tugas akhir ini. Selain itu, diharapkan dapat berguna untuk pembaca yang tertarik untuk melakukan pengembangan lebih lanjut. Secara garis besar, buku tugas akhir terdiri atas beberapa bagian seperti berikut ini.

#### **Bab I Pendahuluan**

Bab yang berisi mengenai latar belakang, rumusan masalah, batasan masalah, tujuan, dan manfaat dari pembuatan tugas akhir. Selain itu metodologi yang digunakan dan sistematika penulisan laporan akhir juga merupakan bagian dari bab ini.

#### **Bab II Tinjauan Pustaka**

Bab ini berisi penjelasan secara detail mengenai dasar-dasar penunjang dan teori-teori yang digunakan untuk mendukung pembuatan tugas akhir ini.

#### **Bab III Analisis dan Perancangan Sistem**

Bab ini berisi tentang analisis permasalahan, deskripsi umum sistem, spesifikasi kebutuhan perangkat lunak, lingkungan perancangan, perancangan arsitektur sistem, diagram kelas, dan struktur data.

#### **Bab IV Implementasi**

Bab ini membahas implementasi dari desain yang telah dibuat pada bab sebelumnya. Penjelasan berupa kode sumber yang digunakan untuk proses implementasi.



**Bab V Pengujian dan Evaluasi**

Bab ini menjelaskan kemampuan perangkat lunak dengan melakukan pengujian kebenaran dan pengujian kinerja dari sistem yang telah dibuat.

**Bab VI Kesimpulan dan Saran**

Bab ini menjelaskan kemampuan perangkat lunak dengan melakukan pengujian kebenaran dan pengujian *kinerja dari sistem yang telah dibuat.*

*[Halaman ini sengaja dikosongkan]*

## **BAB II**

### **TINJAUAN PUSTAKA**

Bab ini menjelaskan teori-teori yang berkaitan dengan pembangunan aplikasi *web* dan *mobile* sistem kehadiran yang diajukan untuk tugas akhir ini. Penjelasan ini bertujuan untuk memberikan gambaran secara umum terhadap perangkat lunak yang dibuat dan berguna sebagai penunjang dalam pengembangan perangkat lunak.

#### **2.1 Penelitian Terkait**

Terdapat penelitian terkait rancang bangun sistem informasi pencatatan kehadiran sebelumnya, yaitu Sistem Informasi Pencatatan Kehadiran Perkuliahan Dosen dan Mahasiswa dengan Mengimplementasikan Teknologi Pemindai Sidik. Penelitian ini merupakan Tugas Akhir yang telah selesai dikerjakan oleh salah satu mahasiswa angkatan 2011 di Jurusan Teknik Informatika dan juga mahasiswa Jurusan Sistem Informasi dengan memindai sidik jarinya pada alat pemindai sidik jari [2] . Data sidik jari kemudian diolah dan disimpan pada server basis data. Sehingga kehadiran mahasiswa dan dosen dapat tersimpan dan terekap di dalam basis data. Pencatatan dengan menggunakan sidik jari memanfaatkan teknologi biometric dimana penggunaannya lebih efektif dan efisien serta menghindari munculnya peluang dalam memanipulasi data pencatatan kehadiran serta dengan dibangunnya sistem informasi pencatatan kehadiran perkuliahan mahasiswa dan dosen lebih memudahkan dalam proses pengelolaannya. Namun sistem pencatatan kehadiran menggunakan sidik jari dirasa kurang fleksibilitas dari penerapan yang ada. Fleksibilitas yang dimaksud adalah teknologi yang digunakan tidak bisa berpindah tempat dan diakses kapanpun serta harganya yang mahal.

Telah banyak juga yang menciptakan aplikasi monitoring siswa atau peserta didik. Ada yang melalui SMS Gateway, Aplikasi Web maupun Aplikasi Mobile. Tujuan dari aplikasi-

aplikasi tersebut adalah agar adanya kemudahan akses informasi mengenai siswa atau peserta didik yang diakses baik oleh pengajar maupun oleh orang tua. Contohnya di Kota Bandung, Dinas Pendidikan kotanya telah meluncurkan aplikasi bagi orang tua untuk memantau perkembangan pendidikan anaknya [3]. Pada aplikasi tersebut, Orang tua bisa mendapatkan informasi tentang kehadiran anak, hasil ujian, bahkan mengetahui perkembangan pendidikan melalui input yang dilakukan oleh sekolah. Namun semua aplikasi tersebut tidak dalam ranah perkuliahan. Maka dirancanglah tugas akhir ini, agar dalam perkuliahan, orang tua tetap dapat mengawasi anaknya.

## **2.2 Teknik Informatika ITS**

Jurusan Teknik Informatika yang awalnya tergabung dalam Fakultas Teknologi Industri (FTI). Mulai tahun 2002 menjadi bagian dari Fakultas Teknologi Informasi (FTIf), sesuai dengan Keputusan Menteri Pendidikan nasional RI Nomor : 109/O/2002, tanggal 2 Juli 2002 tentang Pendirian Fakultas Teknologi Informasi Pada Institut Teknologi Sepuluh Nopember [4].

Jurusan Teknik Informatika menawarkan delapan bidang keahlian, yaitu komputasi cerdas dan visi (KCV); komputasi berbasis jaringan (KBJ); rekayasa perangkat lunak (RPL); algoritma dan pemrograman (AP); dasar dan terapan komputer (DTK); interaksi, grafika dan seni (IGS); manajemen informasi (MI) dan arsitektur dan jaringan komputer (AJK).

Perkuliahan di Teknik Informatika terdiri dari 16 Minggu perkuliahan termasuk UTS dan UAS. Terdapat delapan kelas untuk S-1 dan sepuluh Lab untuk kegiatan perkuliahan.

## **2.3 Sistem Kehadiran**

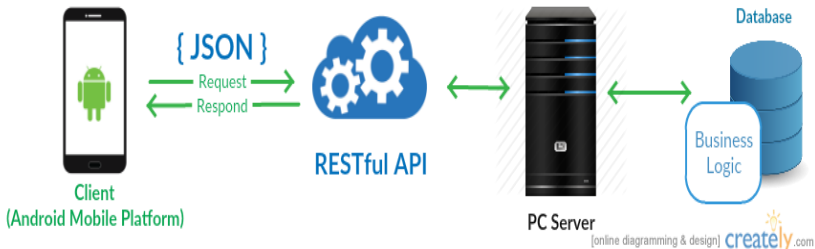
Sistem berasal dari bahasa Latin (systēma) dan bahasa Yunani (sustēma) adalah suatu kesatuan yang terdiri komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi atau energi untuk mencapai suatu tujuan.

Istilah ini sering dipergunakan untuk menggambarkan suatu set entitas yang berinteraksi, di mana suatu model matematika seringkali bisa dibuat [5]. Sedangkan kehadiran menurut KBBI merupakan perihal hadir; adanya (seseorang, sekumpulan orang) pada suatu tempat [6]. Berdasarkan definisi sistem dan kehadiran, dapat disimpulkan bahwa Sistem kehadiran adalah sistem yang digunakan dalam pengambilan data guna mengetahui jumlah kehadiran pada suatu acara. Setiap kegiatan yang membutuhkan informasi mengenai peserta tentu akan melakukan konfirmasi jika telah hadir. Hal ini juga terjadi pada proses kegiatan perkuliahan mahasiswa. Kegunaan sistem kehadiran ini terjadi pada pihak mahasiswa dan pihak dosen maupun kampus. Sistem kehadiran dapat dilakukan dengan berbagai cara, Di Teknik Informatika ITS, Sistem kehadiran mahasiswa dilakukan secara manual yaitu melakukan tanda tangan di kertas daftar hadir mahasiswa. Salah satu kegunaan sistem kehadiran bagi mahasiswa adalah untuk syarat untuk mengikuti ujian mata kuliah, sedangkan bagi pihak dosen atau kampus, sistem kehadiran diperlukan sebagai salah satu bahan evaluasi kepada kepuasan mahasiswa terhadap suatu mata pelajaran dan pembuatan tolak ukur ke depan guna pemberian ilmu yang lebih baik.

## 2.4 Laravel

Laravel adalah framework PHP yang dikembangkan pertama kali oleh Taylor Otwell. Fitur-fitur laravel yang akan dipergunakan pada Aplikasi ini adalah

1. HTTP Response, termasuk JSON Response untuk mengintegrasikan Aplikasi Mobile dengan Web. Digunakan untuk proses *Login* dan pengambilan data dari Database Server. Juga akan dibuat sebuah *web service* untuk menghubungkan antara Aplikasi Mobile dan Server. Aplikasi Mobile akan mengirimkan JSON Request kepada API, yang kemudian akan dikembalikan dalam bentuk JSON Respond seperti pada Gambar 2.1



**Gambar 2.1 Pertukaran Data melalui RESTful API**

2. RestFul Routing, Memudahkan pengguna untuk mengelola request. Karena menggunakan metode standar REST seperti GET, POST, PUT, PATCH, DELETE, STORE. Routing dalam laravel sangat membantu juga untuk *debugging* dan *developing*.
3. Fitur Autentikasi. Laravel telah menyediakan keamanan dalam hal autentikasi, seperti enkripsi, autentikasi *password*, *password reminder* dan *reset*.
4. Fitur Schema Builder untuk berbagai database
5. Fitur Migration & Seeding untuk berbagai database
6. Fitur Query Builder yang *powerful*
7. Eloquent ORM (*Object Relational Mapping*). Untuk mempermudah pemanggilan data melalui model.
8. Scheduling Jobs, Untuk melakukan perintah yang akan dilakukan secara otomatis dan terjadwal [7]
9. Laravel FCM (Firebase Cloud Messaging) untuk pembuatan dan pengiriman *push notification* [8]

## 2.5 JSON

JSON atau *JavaScript Object Notation* adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (generate) oleh

komputer. Format JSON dibuat berdasarkan bagian dari bahasa pemrograman JavaScript [10].

## **2.6 SQLite**

SQLite merupakan sebuah sistem manajemen database relasional. SQLite bersifat open source database. SQLite telah ada cukup lama, cukup stabil, dan sangat terkenal pada perangkat kecil, termasuk Android [11]. SQLite pada aplikasi ini digunakan untuk menyimpan data pada android agar aplikasi tetap dapat menampilkan data disaat tidak tersambung dengan server.

## **2.7 MySQL**

MySQL adalah sebuah perangkat lunak sistem manajemen basis data atau DBMS yang multithread, multi-user. MySQL adalah sebuah implementasi dari sistem manajemen basisdata relasional (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (General Public License).DBMS yang akan digunakan pada aplikasi ini adalah MySQL.[12]

*[Halaman ini sengaja dikosongkan]*



## **BAB III**

### **ANALISIS DAN PERANCANGAN SISTEM**

Bab ini membahas analisis kebutuhan dan rancangan yang akan digunakan untuk membangun perangkat lunak yang diajukan sebagai tugas akhir.

#### **3.1 Analisis**

Tahap analisis dibagi menjadi beberapa bagian, antara lain analisis permasalahan, deskripsi umum sistem, dan spesifikasi kebutuhan perangkat lunak.

##### **3.1.1 Analisis Permasalahan**

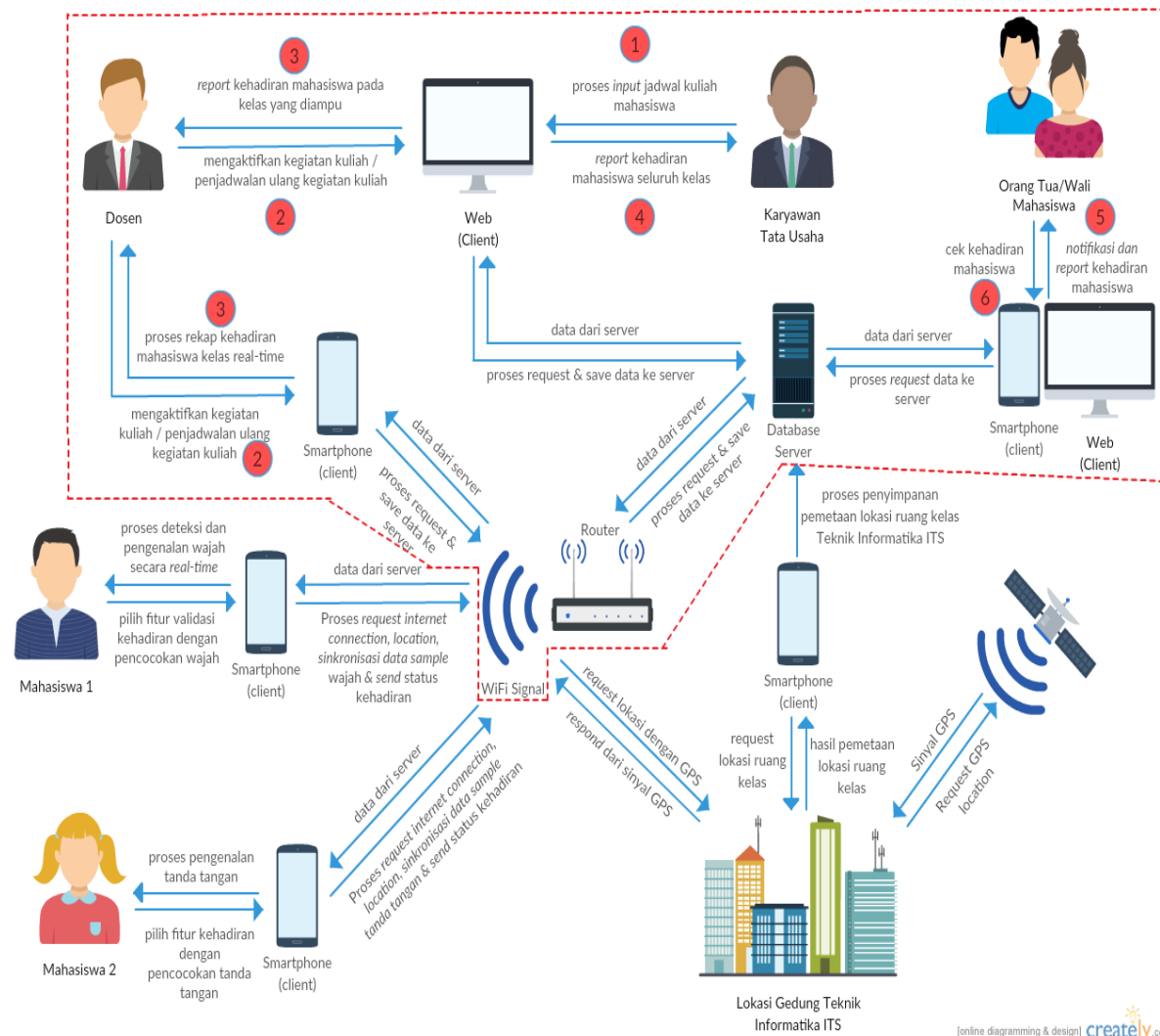
Proses pencatatan kehadiran mahasiswa di Teknik Informatika untuk saat ini masih menggunakan cara manual yaitu menggunakan lembar presensi. Mahasiswa melakukan presensi kehadiran dengan cara tanda tangan di kertas daftar kehadiran. Dalam melakukan presensi kehadiran sering kali terjadi kecurangan yaitu *titip absen*. *Titip absen* merupakan kejadian di mana mahasiswa tercatat hadir di daftar hadir kertas tetapi kenyataannya mahasiswa tersebut tidak hadir di kelas. Jika guru tidak mengawasi presensi kehadiran tidak memungkinkan kecurangan tersebut dapat terjadi. Pencatatan kehadiran mahasiswa yang aman dan efisien sangat diperlukan untuk mengatasi kecurangan presensi kehadiran tersebut. Dan juga perlunya kemudahan bagi Dosen dan Orang Tua untuk memonitor kehadiran mahasiswa dan anaknya pada perkuliahan, agar dapat selalu mengawasi di saat anak tersebut akan melebihi batas ketidakhadiran.

Pada saat ini, perkembangan teknologi informasi khususnya pada teknologi perangkat bergerak dan internet sangat pesat. Akses dan pengambilan informasi dikatakan semakin mudah dan cepat diakses melalui penggunaan perangkat bergerak dan web. Maka pada pengerjaan tugas akhir ini dibangunlah aplikasi *mobile* yang menggantikan proses sistem kehadiran

mahasiswa dari sistem manual melalui tanda tangan pada kertas menjadi sistem *online* pada aplikasi *mobile*. Selain itu, dengan aplikasi ini diharapkan juga dapat memberikan kebenaran data kehadirann (*data validation*) serta mempermudah pemantauan kehadiran mahasiswa di setiap mata kuliah. Selain itu, aplikasi ini juga dapat memberikan informasi kepada orang tua/wali mahasiswa tentang status kehadiran anaknya. Notifikasi akan dikirimkan secara otomatis ke *device* orang tua/wali jika anaknya tidak masuk dalam perkuliahan. Informasi ini diharapkan dapat membantu orang tua/wali mahasiswa dalam memantau anaknya.

Pada gambar gambar 3.1 merupakan gambaran umum dari arsitektur perangkat lunak yang akan dibangun. Pada gambar ditunjukkan cara kerja keseluruhan dari sistem ini. Fokus pengerjaan pada Tugas Akhir ini adalah bagian yang dikelilingi dengan garis merah putus-putus. Perangkat lunak yang dibangun meliputi *website* untuk pengelolaan sistem kehadiran dan perangkat *mobile* untuk input.

### 3.1.2 Deskripsi Umum Sistem



Gambar 3.1 Deskripsi Umum Sistem



Pada awalnya, karyawan Tata Usaha harus melakukan proses *login* ke dalam aplikasi *web* Sistem Kehadiran Mahasiswa terlebih dahulu untuk pengaturan jadwal dan daftar mahasiswa(1). Kemudian pada saat mata kuliah akan berlangsung, Dosen akan melakukan proses *login* ke dalam aplikasi *mobile* atau *web* untuk melakukan pengaktifan mata kuliah yang akan berlangsung(2), atau melakukan penjadwalan ulang(2). Kemudian apabila dosen telah melakukan pengaktifan, Proses pencocokan lokasi diadakannya kuliah dan lokasi mahasiswa akan dilakukan. Apabila lokasi cocok, maka kemudian mahasiswa dapat memilih dua cara untuk melakukan validasi kehadiran. *Face Recognition*, pencocokan tandatangan atau pencocokan suara. Setelah suatu proses validasi absen berhasil, maka secara *real time* status kehadiran mahasiswa akan terupdate pada server, dan dapat dilihat pada Aplikasi *mobile* milik Dosen(3). Dan juga melalui Aplikasi *Web*, Dosen, Karyawan dapat melihat *report* kehadiran perkuliahan(4). Orang tua/wali akan mendapatkan notifikasi apabila anaknya tidak masuk dalam perkuliahan(5). Selain itu juga dapat melihat rekap kehadiran anaknya melalui aplikasi *web*(6)

Sesuai dengan cakupan perangkat lunak yang telah dijelaskan pada bagian deskripsi umum sistem, dibutuhkan adanya spesifikasi perangkat lunak agar dapat memberikan solusi dari permasalahan yang diberikan dan dapat bekerja dengan baik dalam mengakomodasi kebutuhan. Diharapkan dengan adanya spesifikasi ini dapat menyesuaikan kebutuhan-kebutuhan pengguna. Spesifikasi kebutuhan perangkat lunak adalah penjelasan mengenai kebutuhan sistem yang diinginkan pelanggan atau klien dalam bentuk tulisan. Spesifikasi kebutuhan perangkat lunak pada tugas akhir ini terdiri dari kebutuhan fungsional, kebutuhan non-fungsional, aktor, dan kasus penggunaan.

### 3.1.3 Spesifikasi Kebutuhan Perangkat Lunak

#### 3.1.3.1 Kebutuhan Fungsional

**Tabel 3.1 Kebutuhan Fungsional**

<b>No</b>	<b>Kebutuhan Fungsional</b>	<b>Deskripsi</b>
<b>1</b>	Mengelola data mahasiswa	Menampilkan data mahasiswa, melakukan proses penambahan, pengubahan, serta penghapusan data mahasiswa
<b>2</b>	Mengelola data dosen	Menampilkan data dosen, melakukan proses penambahan, pengubahan, serta penghapusan data dosen
<b>3</b>	Mengelola data matakuliah	Menampilkan data matakuliah, melakukan proses penambahan, pengubahan, serta penghapusan data matakuliah
<b>4</b>	Mengelola data ruangan	Menampilkan data ruangan, melakukan proses penambahan, pengubahan, serta penghapusan data ruangan
<b>5</b>	Mengelola data kelas	Menampilkan data kelas, melakukan proses penambahan, pengubahan, serta penghapusan data kelas
<b>6</b>	Mengelola data perkuliahan	Menampilkan data perkuliahan, dan pengubahan data perkuliahan
<b>7</b>	Menagelola data kehadiran mahasiswa	Menampilkan data kehadiran mahasiswa, pengubahan, perekapan data kehadiran mahasiswa
<b>8</b>	Mengelola data orang tua	Menampilkan data orang tua, melakukan proses penambahan, pengubahan, serta penghapusan data orang tua
<b>9</b>	Mengelola wajah dan tanda tangan	Melakukan upload data wajah dan tanda tangan ke server,

		pengambilan data wajah dan Tanda tangan dari server

### 3.1.3.2 Kebutuhan Non-Fungsional

Kebutuhan non-fungsional yang harus dipenuhi oleh sistem sebagai berikut:

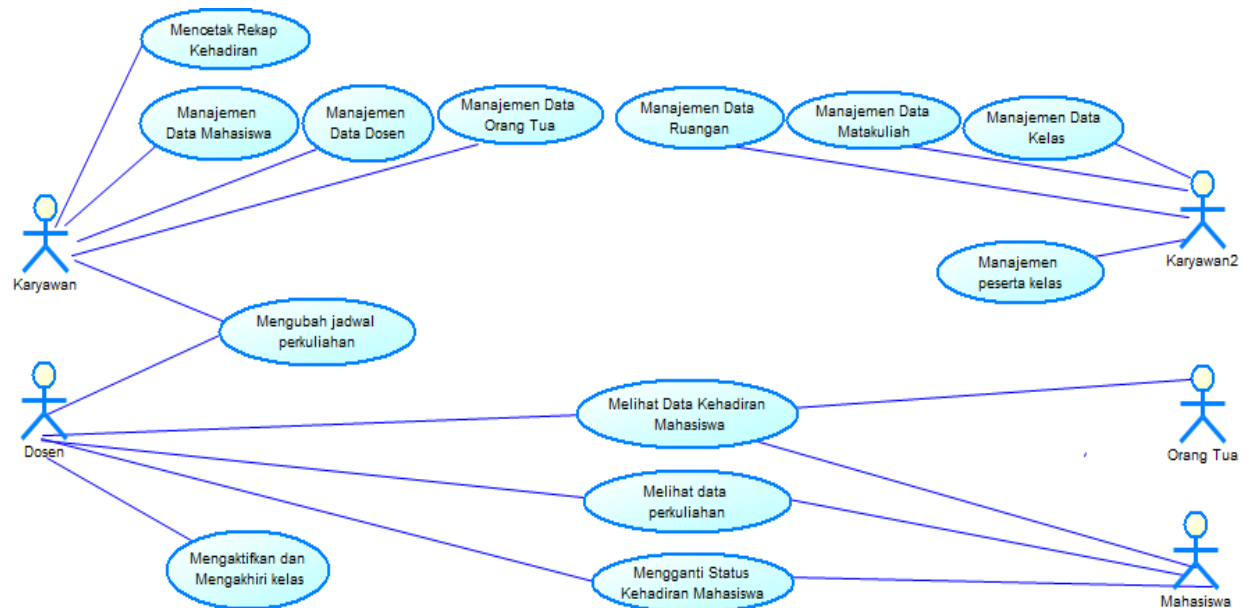
1. Kebutuhan Performa  
Perangkat lunak akan berjalan dengan performa terbaik jika dijalankan di atas spesifikasi minimal.
2. Kebutuhan Perlindungan Keamanan  
*Username* dan *password* dibutuhkan untuk mengamankan data pengguna yang berhak mengakses sistem sebagai syarat memasuki sistem dan melakukan semua fungsionalitas pada sistem.

### 3.1.3.3 Aktor

Pengertian pengguna adalah pihak-pihak, baik manusia maupun sistem atau perangkat lain yang terlibat dan berinteraksi secara langsung dengan sistem. Pada perangkat lunak ini terdapat tiga pengguna untuk aplikasi *web* yaitu dosen, karyawan dan orang tua. Sedangkan pada aplikasi *mobile* terdapat tiga pengguna, yakni, mahasiswa, dosen dan orang tua.

### 3.1.3.4 Kasus Penggunaan

Berdasarkan analisis spesifikasi kebutuhan fungsional dan analisis aktor dari sistem, dibuat kasus penggunaan sistem. Kasus penggunaan digambarkan dalam tabel penjelasan kasus penggunaan dan diagram kasus penggunaan. Tabel penjelasan kasus penggunaan dapat dilihat pada Tabel 3.2 dan diagram kasus penggunaan dapat dilihat pada Gambar 3.2



**Gambar 3.2 Diagram Kasus Penggunaan**



**Tabel 3.2 Kasus Penggunaan**

<b>Kode Kasus Penggunaan</b>	<b>Nama</b>	<b>Aktor</b>
UC-001	Melihat data mahassiswa	Karyawan
UC-002	Menambah data mahasiswa	Karyawan
UC-003	Mengubah data mahasiswa	Karyawan, Mahasiswa
UC-004	Menghapus data mahasiswa	Karyawan
UC-005	Melihat data dosen	Karyawan
UC-006	Menambah data dosen	Karyawan
UC-007	Mengubah data dosen	Karyawan, Dosen
UC-008	Menghapus dosen	Karyawan
UC-009	Melihat data matakuliah	Karyawan
UC-010	Menambah data matakuliah	Karyawan
UC-011	Mengubah data matakuliah	Karyawan
UC-012	Menghapus data matakuliah	Karyawan
UC-013	Melihat data ruangan	Karyawan
UC-014	Menambah data ruangan	Karyawan
UC-015	Mengubah data ruangan	Karyawan
UC-016	Menghapus data ruangan	Karyawan
UC-017	Melihat data kelas	Karyawan, Dosen, Mahasiswa
UC-018	Menambah data kelas	Karyawan
UC-019	Mengubah data kelas	Karyawan
UC-020	Menghapus data kelas	Karyawan
UC-021	Menampilkan data perkuliahan	Karyawan, Dosen, Orang

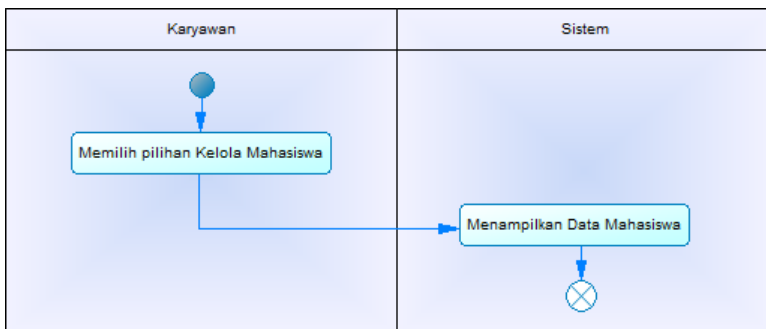
		Tua
UC-022	Mengubah jadwal perkuliahan	Karyawan, Dosen
UC-023	Menampilkan data kehadiran mahasiswa	Karyawan, Dosen, Mahasiswa
UC-024	Mengubah status kehadiran mahasiswa	Karyawan, Dosen,
UC-025	Melihat data orangtua	Karyawan
UC-026	Menambah data orangtua	Karyawan
UC-027	Mengubah data orangtua	Karyawan
UC-028	Menghapus data orangtua	Karyawan
UC-029	Menambah peserta kelas	Karyawan
UC-030	Menghapus peserta kelas	Karyawan
UC-031	Melihat data waktuperkuliahan	Karyawan
UC-032	Menambah data waktuperkuliahan	Karyawan
UC-033	Mengubah data waktuperkuliahan	Karyawan
UC-034	Menghapus data waktuperkuliahan	Karyawan
UC-035	Menerima Notifikasi	Dosen, Mahasiswa, Orang Tua
UC-036	Mengubah Status Perkuliahan	Dosen

### 3.1.3.4.1 Melihat Data Mahasiswa (UC-001)

Kasus penggunaan nomor UC-001 ini diakses ketika Karyawan hendak melihat data mahasiswa yang ada pada sistem. Spesifikasi, dan diagram aktivitas, dan sekuens kasus penggunaan ini dapat dilihat pada Tabel 3.3, , dan Gambar 3.3

**Tabel 3.3 Spesifikasi Kasus Penggunaan UC-001**

<b>Kode Use Case</b>	UC-0001	
<b>Nama Use Case</b>	Melihat data mahasiswa	
<b>Aktor</b>	Karyawan	
<b>Deskripsi</b>	Karyawan dapat melihat data mahasiswa	
<b>Relasi</b>	-	
<b>Kondisi Awal</b>	Sistem belum menampilkan data mahasiswa	
<b>Kondisi Akhir</b>	Sistem sudah menampilkan data mahasiswa	
<b>Alur kejadian normal</b>	<b>Aktor</b>	<b>Sistem</b>
	1. Memilih Kelola Mahasiswa	
		2. Menampilkan Data Mahasiswa



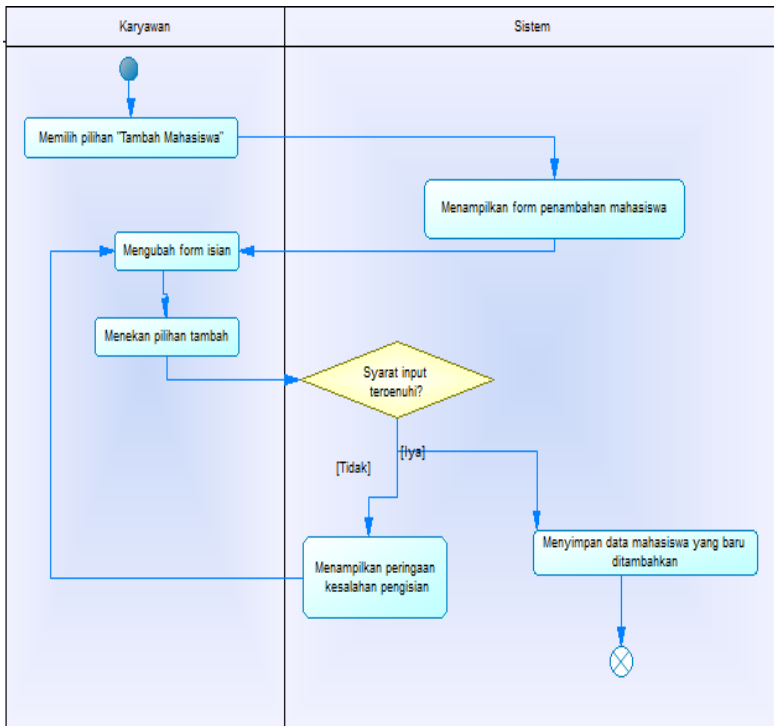
**Gambar 3.3 Diagram Aktifitas Melihat Data Mahasiswa**

### 3.1.3.4.2 Menambah Data Mahasiswa (UC-002)

Kasus penggunaan nomor UC-002 ini diakses ketika karyawan hendak menambah data Mahasiswa pada sistem. Untuk menambah Mahasiswa, sistem akan menampilkan form input NRP, Nama, Alamat, Email, No Telp dan Tanggal Lahir. Spesifikasi dan diagram aktivitas kasus penggunaan ini dapat dilihat pada Tabel 3.4, Gambar 3.4

**Tabel 3.4 Spesifikasi Kasus Penggunaan UC-002**

<b>Kode Use Case</b>	UC-0002	
<b>Nama Use Case</b>	Menambah data mahasiswa	
<b>Aktor</b>	Karyawan	
<b>Deskripsi</b>	Karyawan dapat menambah data mahasiswa pada sistem	
<b>Relasi</b>	-	
<b>Kondisi Awal</b>	Karyawan belum menambahkan data mahasiswa yang baru	
<b>Kondisi Akhir</b>	Sistem sudah menyimpan data mahasiswa yang ditambah oleh karyawan	
<b>Alur kejadian normal</b>	<b>Aktor</b>	<b>Sistem</b>
	1. Memilih Tambah Mahasiswa	
		2. Menampilkan form penambahan mahasiswa
	3. Mengisi Form	
	4. Memilih pilihan kirim	
		5a. Menyimpan data mahasiswa
<b>Alur Kejadian Alternatif</b>		5b Menampilkan peringatan kesalahan pengisian



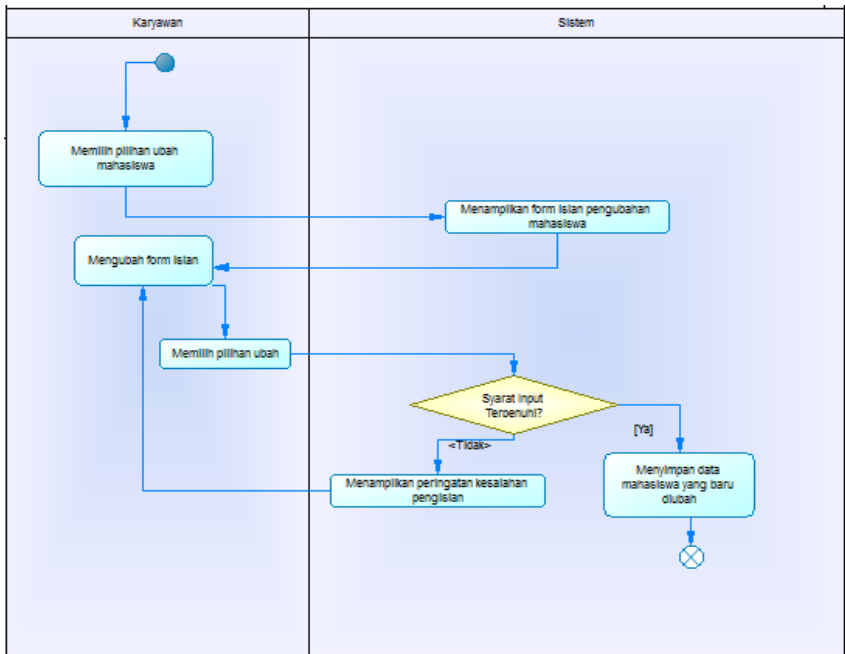
**Gambar 3.4 Diagram Aktivitas Menambah Data Mahasiswa**

#### 3.1.3.4.3 Mengubah Data Mahasiswa (UC-003)

Kasus penggunaan nomor UC-003 ini diakses ketika karyawan hendak merubah data seorang mahasiswa. Pada kasus penggunaan ini, sistem menampilkan form untuk pengisian perubahan data pada mahasiswa yang ingin dirubah. Spesifikasi, diagram aktivitas, dan sekuens kasus penggunaan ini dapat dilihat pada Tabel 3.5 dan Gambar 3.5

**Tabel 3.5 Spesifikasi Kasus Penggunaan UC-003**

<b>Kode Use Case</b>	UC-0003	
<b>Nama Use Case</b>	Mengubah data mahasiswa	
<b>Aktor</b>	Karyawan	
<b>Deskripsi</b>	Karyawan dapat mengubah data mahasiswa	
<b>Relasi</b>	-	
<b>Kondisi Awal</b>	Karyawan belum mengubah data mahasiswa yang perlu diubah	
<b>Kondisi Akhir</b>	Sistem sudah menyimpan data mahasiswa yang diubah	
<b>Alur kejadian normal</b>	<b>Aktor</b>	<b>Sistem</b>
	1. Memilih pilihan ubah mahasiswa	
		2. Menampilkan form isian perubahan
	3. Mengubah form isian	
	4. Memilih pilihan ubah	
<b>Alur kejadian alternative</b>		5a. Menyimpan perubahan data
	<b>Aktor</b>	<b>Sistem</b>
		5b. Menampilkan peringatan kesalahan pengisian



**Gambar 3.5 Diagram Aktivitas Mengubah Data Mahasiswa**

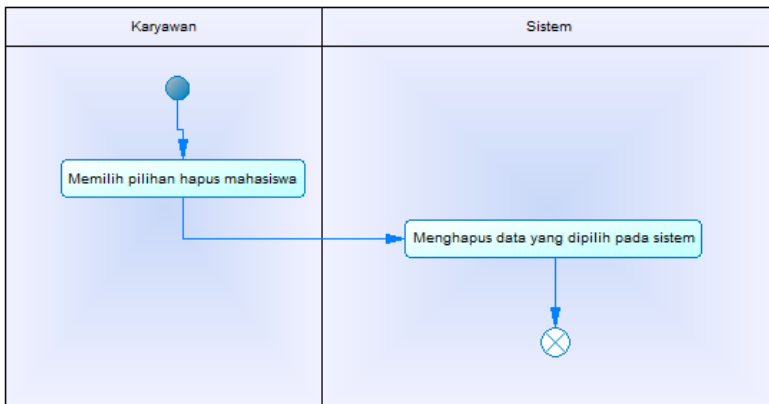
#### 3.1.3.4.4 Menghapus Data Mahasiswa (UC-004)

Kasus penggunaan nomor UC-004 ini dilakukan karyawan ketika akan menghapus data mahasiswa. Spesifikasi, diagram aktivitas, dan sekuens kasus penggunaan ini dapat dilihat pada Tabel 3.6, Gambar 3.6

**Tabel 3.6 Spesifikasi Kasus Penggunaan UC-004**

<b>Kode Use Case</b>	UC-0004
<b>Nama Use Case</b>	Menghapus data mahasiswa
<b>Aktor</b>	Karyawan
<b>Deskripsi</b>	Karyawan dapat menghapus data mahasiswa
<b>Relasi</b>	-
<b>Kondisi Awal</b>	Karyawan belum menghapus data mahasiswa yang dimaksud

<b>Kondisi Akhir</b>	Sistem sudah menghapus data mahasiswa yang dihapus oleh karyawan	
<b>Alur kejadian normal</b>	<b>Aktor</b>	<b>Sistem</b>
	1. Memilih pilihan hapus mahasiswa	
		2. Menghapus data yang sudah dipilih pada sistem
<b>Alur kejadian alternative</b>	<b>Aktor</b>	<b>Sistem</b>



**Gambar 3.6 Diagram Aktivitas Menghapus Data Mahasiswa**

### 3.1.3.4.5 Melihat Data Dosen (UC-005)

Kasus penggunaan nomor UC-005 ini sama dengan kasus penggunaan nomor UC-001. Namun data yang ditampilkan disesuaikan dengan tabel dosen



#### **3.1.3.4.6 Menambah Data Dosen (UC-006)**

Kasus penggunaan nomor UC-006 ini sama dengan kasus penggunaan nomor UC-003. Namun form input yang ditampilkan disesuaikan dengan tabel dosen.

#### **3.1.3.4.7 Mengubah Data Dosen (UC-006)**

Kasus penggunaan nomor UC-007 ini sama dengan kasus penggunaan nomor UC-003. Namun form input yang ditampilkan disesuaikan dengan tabel dosen.

#### **3.1.3.4.8 Menghapus Data Dosen (UC-008)**

Kasus penggunaan nomor UC-008 ini sama dengan kasus penggunaan nomor UC-004. Namun yang dihapus adalah data dosen.

#### **3.1.3.4.9 Melihat Data Matakuliah(UC-009)**

Kasus penggunaan nomor UC-009 ini sama dengan kasus penggunaan nomor UC-001. Namun data yang ditampilkan disesuaikan dengan tabel matakuliah

#### **3.1.3.4.10 Menambah Data Matakuliah (UC-010)**

Kasus penggunaan nomor UC-010 ini sama dengan kasus penggunaan nomor UC-003. Namun form input yang ditampilkan disesuaikan dengan tabel matakuliah.

#### **3.1.3.4.11 Mengubah Data Matakuliah (UC-011)**

Kasus penggunaan nomor UC-00011 ini sama dengan kasus penggunaan nomor UC-0003. Namun form input yang ditampilkan disesuaikan dengan tabel matakuliah.

**3.1.3.4.12 Menghapus Data Matakuliah(UC-012)**

Kasus penggunaan nomor UC-012 ini sama dengan kasus penggunaan nomor UC-004. Namun yang dihapus adalah data matakuliah.

**3.1.3.4.13 Melihat Data Ruangan (UC-013)**

Kasus penggunaan nomor UC-013 ini sama dengan kasus penggunaan nomor UC-001. Namun data yang ditampilkan disesuaikan dengan tabel ruangan.

**3.1.3.4.14 Menambah Data Ruangan (UC-014)**

Kasus penggunaan nomor UC-014 ini sama dengan kasus penggunaan nomor UC-003. Namun form input yang ditampilkan disesuaikan dengan table ruangan.

**3.1.3.4.15 Mengubah Data Ruangan (UC-015)**

Kasus penggunaan nomor UC-0015 ini sama dengan kasus penggunaan nomor UC-003. Namun form input yang ditampilkan disesuaikan dengan tabel ruangan.

**3.1.3.4.16 Menghapus Data Ruangan(UC-016)**

Kasus penggunaan nomor UC-016 ini sama dengan kasus penggunaan nomor UC-004. Namun yang dihapus adalah data ruangan.

**3.1.3.4.17 Melihat Data Kelas (UC-017)**

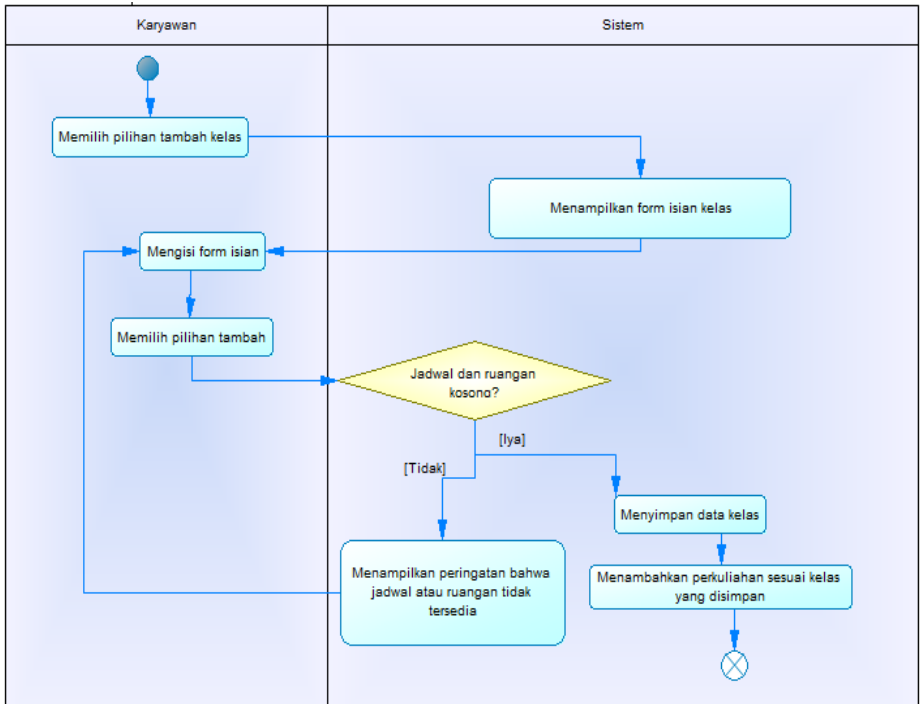
Kasus penggunaan nomor UC-017 ini sama dengan kasus penggunaan nomor UC-001. Namun data yang ditampilkan disesuaikan dengan tabel kelas.

### 3.1.3.4.18 Menambah Data Kelas(UC-018)

Kasus penggunaan nomor UC-018 ini diakses ketika diakses ketika karyawan hendak menambah data sebuah kelas. Pada kasus penggunaan ini, sistem menampilkan form untuk pengisian data kelas. Spesifikasi dan diagram aktivitas, kasus penggunaan ini dapat dilihat pada Tabel 3.7, Gambar 3.7

**Tabel 3.7 Spesifikasi Kasus Penggunaan UC-0018**

<b>Kode Use Case</b>	UC-018	
<b>Nama Use Case</b>	Menambah data kelas	
<b>Aktor</b>	Karyawan	
<b>Deskripsi</b>	Karyawan dapat menambah data kelas pada sistem	
<b>Relasi</b>	-	
<b>Kondisi Awal</b>	Karyawan belum menambahkan data kelas yang baru	
<b>Kondisi Akhir</b>	Sistem sudah menyimpan data kelas	
<b>Alur kejadian normal</b>	<b>Aktor</b>	<b>Sistem</b>
	1. Memilih tambah kelas	
		2. Menampilkan form isian kelas
	3. Mengisi form isian	
	4. Memilih pilihan tambah kelas	
		5a.Menyimpan data kelas
<b>Alur kejadian alternatif</b>		6 Menambahkan perkuliahan sesuai kelas yang disimpan
	<b>Aktor</b>	<b>Sistem</b>
		5b. Menampilkan peringatan bahwa jadwal dan atau ruangan telah terisi



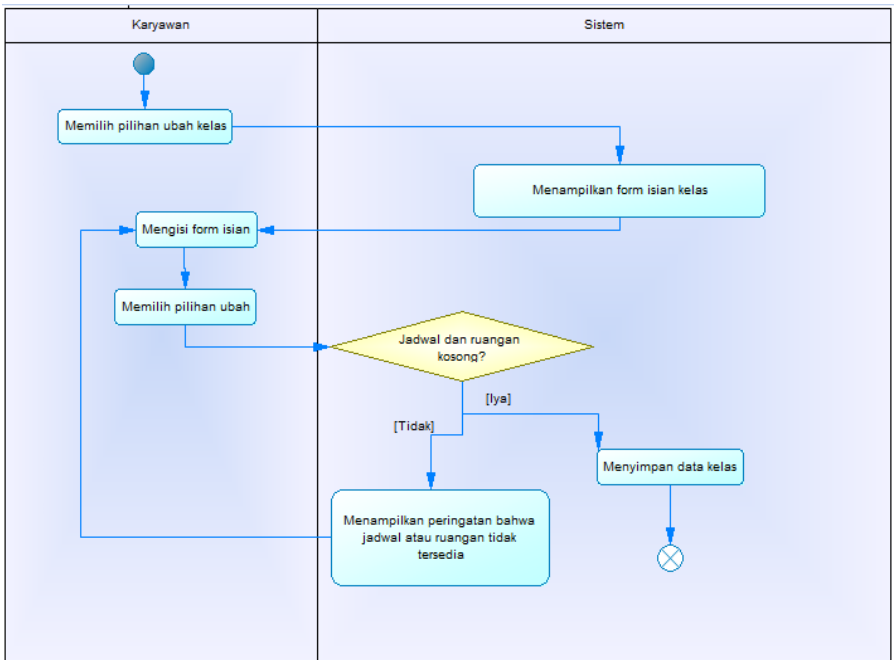
**Gambar 3.7 Diagram Aktivitas Menambah Kelas**

#### 3.1.3.4.19 Mengubah Data Kelas(UC-019)

Kasus penggunaan nomor UC-019 ini diakses ketika diakses ketika karyawan hendak merubah data sebuah kelas. Pada kasus penggunaan ini, sistem menampilkan form untuk pengisian perubahan data pada kelas yang ingin diubah. Spesifikasi, dan diagram aktivitas, kasus penggunaan ini dapat dilihat pada Tabel 3.8 dan Gambar 3.8

**Tabel 3.8 Spesifikasi Kasus Penggunaan UC-019**

<b>Kode Use Case</b>	UC-019	
<b>Nama Use Case</b>	Mengubah data kelas	
<b>Aktor</b>	Karyawan	
<b>Deskripsi</b>	Karyawan dapat mengubah data kelas	
<b>Relasi</b>	-	
<b>Kondisi Awal</b>	Karyawan belum mengubah data kelas yang perlu diedit	
<b>Kondisi Akhir</b>	Sistem sudah menyimpan data kelas yang telah diubah	
<b>Alur kejadian normal</b>	<b>Aktor</b>	<b>Sistem</b>
	1. Memilih pilihan Ubah Kelas	
		2. Menampilkan form isian kelas
	3. Mengisi form isian	
	4. Memilih pilihan ubah	
		4a. Menyimpan perubahan kelas
<b>Alur kejadian alternatif</b>	<b>Aktor</b>	<b>Sistem</b>
		4b. Menampilkan peringatan bahwa jadwal dan atau ruangan telah terisi



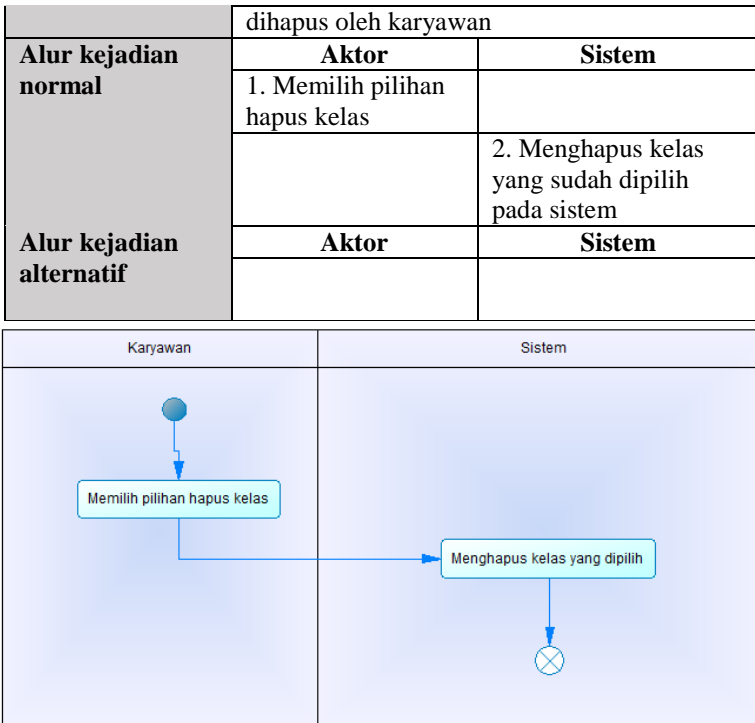
**Gambar 3.8 Diagram Aktivitas Mengubah Kelas**

#### 3.1.3.4.20 Menghapus Data Kelas (UC-020)

Kasus penggunaan nomor UC-020 ini diakses ketika karyawan hendak menghapus data kelas pada sistem. Spesifikasi, dan diagram aktivitas, kasus penggunaan ini dapat dilihat pada Gambar 3.9 dan Gambar 3.9

**Tabel 3.9 Spesifikasi Kasus Penggunaan UC-021**

<b>Kode Use Case</b>	UC-020
<b>Nama Use Case</b>	Menghapus Data Kelas
<b>Aktor</b>	Karyawan
<b>Deskripsi</b>	Karyawan dapat menghapus data kelas
<b>Relasi</b>	-
<b>Kondisi Awal</b>	Karyawan belum menghapus data kelas yang dimaksud
<b>Kondisi Akhir</b>	Sistem sudah menghapus data kelas yang



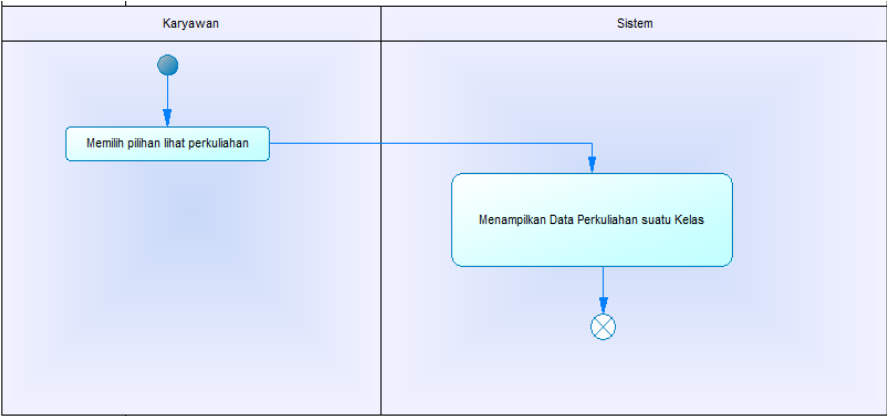
**Gambar 3.9 Diagram Aktivitas Menghapus Kelas**

#### 3.1.3.4.21 Menampilkan Data Perkuliahan (UC-021)

Kasus penggunaan nomor UC-021 ini diakses ketika karyawan atau dosen hendak melihat data perkuliahan pada kelas tertentu pada sistem. Spesifikasi, diagram aktivitas dan sekuens kasus penggunaan ini dapat dilihat pada Tabel 3.10 Gambar 3.10 dan Gambar 0.1

Tabel 3.10 Spesifikasi Kasus Penggunaan UC-021

Kode Use Case	UC-021	
Nama Use Case	Menampilkan data Perkuliahan	
Aktor	Karyawan, Dosen, Orang Tua, Mahasiswa	
Deskripsi	Karyawan, Dosen dan Orang Tua dapat melihat data perkuliahan suatu kelas	
Relasi	-	
Kondisi Awal	Sistem belum menampilkan data perkuliahan suatu kelas	
Kondisi Akhir	Sistem sudah menampilkan data perkuliahan suatu kelas	
Alur kejadian normal	Aktor	Sistem
	1. Memilih pilihan lihat perkuliahan	
		2. Menampilkan Data Perkuliahan suatu Kelas
Alur kejadian alternatif	Aktor	Sistem



Gambar 3.10 Diagram Aktivitas Melihat Perkuliahan



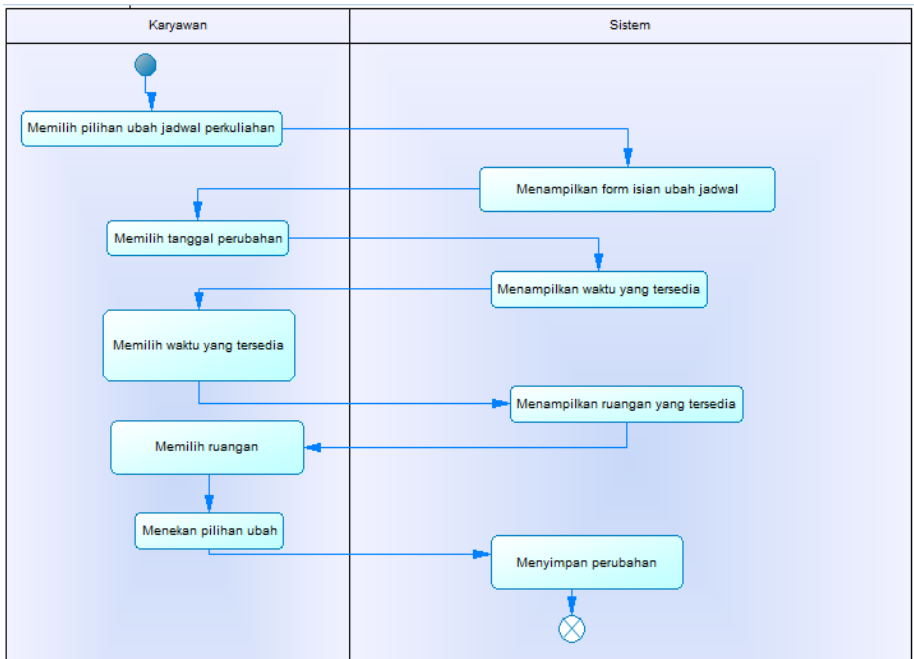
### 3.1.3.4.22 Mengubah Jadwal Perkuliahan (UC-022)

Kasus penggunaan nomor UC-022 ini diakses ketika karyawan atau dosen hendak merubah jadwal suatu perkuliahan. Pada kasus penggunaan ini, Pada kasus penggunaan ini, sistem menampilkan form untuk pengisian perubahan jadwal pada perkuliahan yang dimaksud.. Spesifikasi, diagram aktivitas, dan sekuens kasus penggunaan ini dapat dilihat pada Tabel 3.11, dan Gambar 3.11 Gambar 0.2

**Tabel 3.11 Spesifikasi Kasus Penggunaan UC-0022**

<b>Kode Use Case</b>	UC-022	
<b>Nama Use Case</b>	Mengubah jadwal perkuliahan	
<b>Aktor</b>	Karyawan, Dosen	
<b>Deskripsi</b>	Karyawan atau Dosen dapat mengubah jadwal suatu perkuliahan	
<b>Relasi</b>	-	
<b>Kondisi Awal</b>	Karyawan atau Dosen belum mengubah jadwal suatu perkuliahan	
<b>Kondisi Akhir</b>	Sistem sudah menyimpan jadwal suatu perkuliahan yang diubah	
<b>Alur kejadian normal</b>	<b>Aktor</b>	<b>Sistem</b>
	1. Memilih pilihan ubah jadwal perkuliahan	
		2. Menampilkan form isian ubah jadwal
	3. Memilih tanggal perubahan	
		4. Menampilkan waktu yang tersedia
	5. Memilih waktu perkuliahan	
		6. Menampilkan ruangan yang tersedia
	8 Memilih ruangan	

	9 Memilih pilihan ubah	
		10 Menyimpan perubahan
<b>Alur kejadian alternatif</b>	<b>Aktor</b>	<b>Sistem</b>



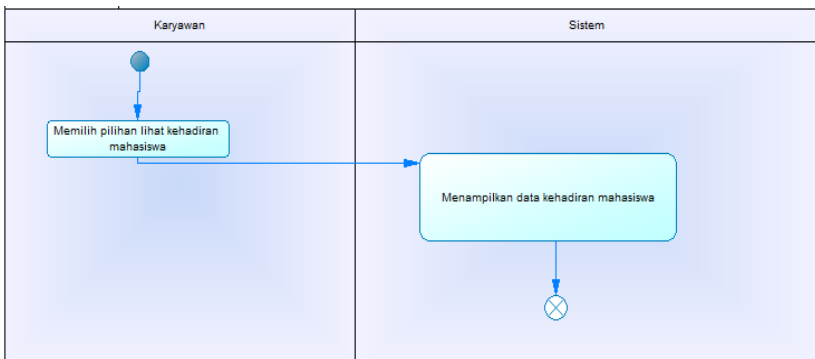
**Gambar 3.11 Diagram Aktivitas Mengubah Jadwal Perkuliahan**

#### 3.1.3.4.23 Menampilkan data kehadiran mahasiswa (UC-023)

Kasus penggunaan nomor UC-023 ini diakses ketika karyawan, dosen dan orang tua akan melihat data kehadiran mahasiswa di suatu kelas. Spesifikasi, diagram aktivitas kasus penggunaan dan Sekuens ini dapat dilihat pada Tabel 3.12 Gambar 3.12 dan Gambar 0.3

**Tabel 3.12 Spesifikasi Kasus Penggunaan UC-023**

<b>Kode Use Case</b>	UC-023	
<b>Nama Use Case</b>	Menampilkan data kehadiran mahasiswa	
<b>Aktor</b>	Karyawan, Dosen, Orang Tua	
<b>Deskripsi</b>	Karyawan, Dosen dan Orang Tua dapat melihat kehadiran mahasiswa di suatu kelas	
<b>Relasi</b>	-	
<b>Kondisi Awal</b>	Sistem belum menampilkan kehadiran mahasiswa	
<b>Kondisi Akhir</b>	Sistem sudah menampilkan kehadiran mahasiswa sesuai kelas yang dimaksud	
<b>Alur kejadian normal</b>	<b>Aktor</b>	<b>Sistem</b>
	1. Memilih pilihan lihat kehadiran mahasiswa	
<b>Alur kejadian alternatif</b>		2. Menampilkan data kehadiran mahasiswa
	<b>Aktor</b>	<b>Sistem</b>

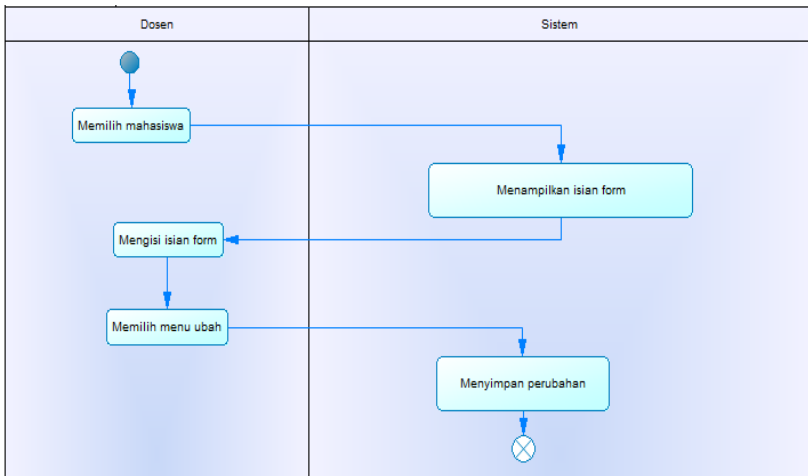
**Gambar 3.12 Diagram Aktivitas Melihat Data Kehadiran**

#### 3.1.3.4.24 Mengisi status kehadiran mahasiswa(UC-024)

Kasus penggunaan nomor UC-024 ini diakses ketika Dosen dan Mahasiswa hendak mengisi status kehadiran mahasiswa di suatu perkuliahan. Spesifikasi, diagram aktivitas, kasus penggunaan dan Sekuens ini dapat dilihat pada Tabel 3.13 Gambar 3.13 dan Gambar 0.4

**Tabel 3.13 Spesifikasi Kasus Penggunaan UC-024**

<b>Kode Use Case</b>	UC-024	
<b>Nama Use Case</b>	Mengubah status kehadiran mahasiswa	
<b>Aktor</b>	Dosen	
<b>Deskripsi</b>	Dosen dapat mengubah status kehadiran mahasiswa di suatu perkuliahan	
<b>Relasi</b>	-	
<b>Kondisi Awal</b>	Sistem telah menampilkan halaman rekap kehadiran mahasiswa	
<b>Kondisi Akhir</b>	Sistem menyimpan status kehadiran	
<b>Alur kejadian normal</b>	<b>Aktor</b>	<b>Sistem</b>
	1. Memilih mahasiswa	
		2.Menampilkan isian form
	3. Mengisi form isian	
		4.Menyimpan perubahan
<b>Alur kejadian alternatif</b>	<b>Aktor</b>	<b>Sistem</b>



**Gambar 3.13 Diagram Aktivitas Mengisi Status Kehadiran**

#### **3.1.3.4.25 Melihat Data Orang Tua (UC-025)**

Kasus penggunaan nomor UC-025 ini sama dengan kasus penggunaan nomor UC-001. Namun data yang ditampilkan disesuaikan dengan tabel orangtua

#### **3.1.3.4.26 Menambah Data Orang Tua (UC-026)**

Kasus penggunaan nomor UC-026 ini sama dengan kasus penggunaan nomor UC-003. Namun form input yang ditampilkan disesuaikan dengan table orangtua.

#### **3.1.3.4.27 Mengubah Data Orang Tua (UC-027)**

Kasus penggunaan nomor UC-0027 ini sama dengan kasus penggunaan nomor UC-003. Namun form input yang ditampilkan disesuaikan dengan tabel orangtua

#### **3.1.3.4.28 Menghapus Data Orang Tua (UC-0028)**

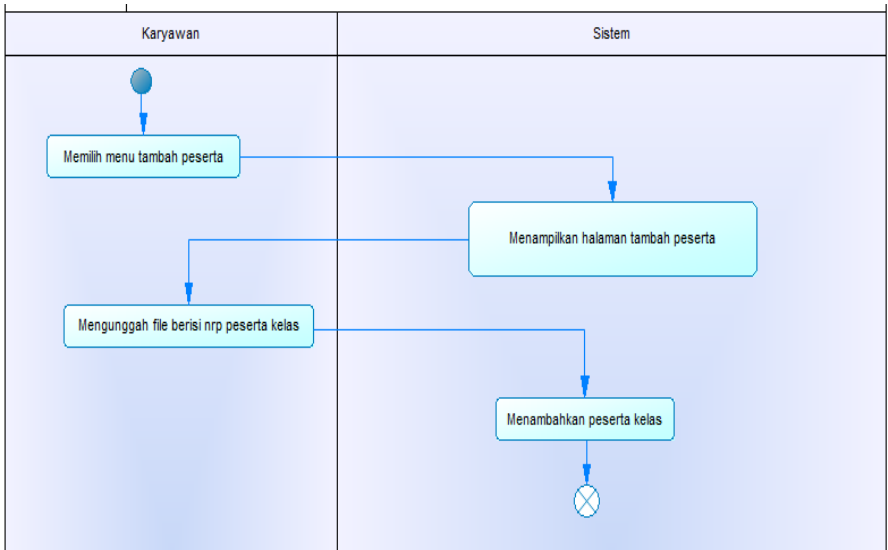
Kasus penggunaan nomor UC-0028 ini sama dengan kasus penggunaan nomor UC-0004. Namun yang dihapus adalah data orangtua

### 3.1.3.4.29 Menambah Peserta Kelas (UC-029)

Kasus penggunaan nomor UC-029 ini adalah saat karyawan hendak menambah peserta kelas tertentu. Inputnya berupa file Excel berisi list nrp. Spesifikasi, diagram aktivitas, dan sekuens kasus penggunaan ini dapat dilihat pada Tabel 3.14 dan Gambar 3.14

**Tabel 3.14 Spesifikasi Kasus Penggunaan UC-029**

<b>Kode Use Case</b>	UC-0029	
<b>Nama Use Case</b>	Menambah Peserta Kelas	
<b>Aktor</b>	Karyawan	
<b>Deskripsi</b>	Karyawan berada pada halaman kelas	
<b>Relasi</b>	-	
<b>Kondisi Awal</b>	Karyawan belum memasukkan peserta kelas	
<b>Kondisi Akhir</b>	Sistem menyimpan peserta kelas	
<b>Alur kejadian normal</b>	<b>Aktor</b>	<b>Sistem</b>
	1. Memilih menu tambah peserta	
		2. Menampilkan halaman tambah peserta kelas
	3. Mengunggah <i>file</i> berisi nrp peserta kelas	
		4. Menambahkan peserta kelas



**Gambar 3.14 Diagram Aktivitas Menambah Peserta Kelas**

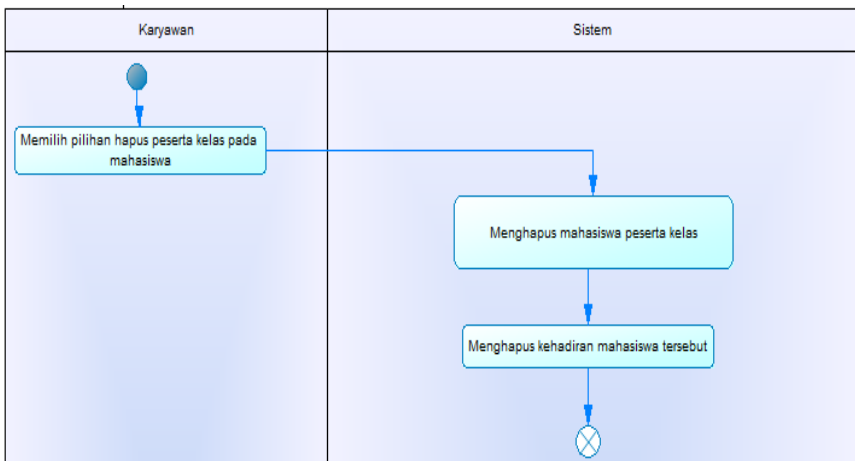
#### **3.1.3.4.30 Menghapus Peserta Kelas(UC-030)**

Kasus penggunaan nomor UC-030 ini adalah saat karyawan hendak menghapus peserta kelas tertentu.. Spesifikasi, dan diagram aktivitas, kasus penggunaan ini dapat dilihat pada Tabel 3.15 dan Gambar 3.15

**Tabel 3.15 Spesifikasi Kasus Penggunaan UC-0030**

<b>Kode Use Case</b>	UC-0030
<b>Nama Use Case</b>	Menghapus Peserta Kelas
<b>Aktor</b>	Karyawan
<b>Deskripsi</b>	Karyawan dapat menghapus mahasiswa pada suatu kelas

<b>Relasi</b>	-	
<b>Kondisi Awal</b>	Karyawan belum menghapus peserta kelas	
<b>Kondisi Akhir</b>	Sistem menghapus peserta kelas sesuai pilihan karyawan	
<b>Alur kejadian normal</b>	<b>Aktor</b>	<b>Sistem</b>
	1. Memilih pilihan hapus peserta kelas pada mahasiswa	
		2. Menghapus mahasiswa peserta kelas.
<b>Alur kejadian normal</b>		3. Menghapus daftar kehadiran mahasiswa tersebut
	<b>Aktor</b>	<b>Sistem</b>



**Gambar 3.15 Diagram Aktivitas Menghapus Peserta Kelas**



#### **3.1.3.4.31 Melihat Data Waktuperkuliah (UC-031)**

Kasus penggunaan nomor UC-031 ini sama dengan kasus penggunaan nomor UC-001. Namun data yang ditampilkan disesuaikan dengan tabel waktuperkuliah

#### **3.1.3.4.32 Menambah Data Waktuperkuliah (UC-032)**

Kasus penggunaan nomor UC-006 ini sama dengan kasus penggunaan nomor UC-003. Namun form input yang ditampilkan disesuaikan dengan tabel waktuperkuliah

#### **3.1.3.4.33 Mengubah Data Waktuperkuliah (UC-033)**

Kasus penggunaan nomor UC-033 ini sama dengan kasus penggunaan nomor UC-003. Namun form input yang ditampilkan disesuaikan dengan tabel waktuperkuliah.

#### **3.1.3.4.34 Menghapus Data Waktuperkuliah (UC-034)**

Kasus penggunaan nomor UC-034 ini sama dengan kasus penggunaan nomor UC-004. Namun yang dihapus adalah data waktuperkuliah.

#### **3.1.3.4.35 Menerima Notifikasi (UC-035)**

Kasus penggunaan nomor UC-035 ini adalah saat Dosen, Orang Tua dan Mahasiswa menerima *push notification* Android melalui Firebase Cloud Messaging. Dosen dan mahasiswa mendapatkan notifikasi 30 menit sebelum perkuliahan dimulai. Kemudian jika ada pergantian jadwal perkuliahan, Mahasiswa akan mendapat notifikasi. Lalu saat mahasiswa absen atau ijin pada perkuliahan, notifikasi akan dikirimkan kepada orang tua mahasiswa tersebut.. Spesifikasi dapat dilihat pada Tabel 3.16

**Tabel 3.16 Spesifikasi Kasus Penggunaan UC-035**

<b>Kode Use Case</b>	UC-0035	
<b>Nama Use Case</b>	Menerima Notifikasi	
<b>Aktor</b>	Dosen, Mahasiswa, Orang Tua	
<b>Deskripsi</b>	Dosen, Mahasiswa, Orang Tua dapat	
<b>Relasi</b>	-	
<b>Kondisi Awal</b>	Aktor sudah terlogin ke dalam sistem	
<b>Kondisi Akhir</b>	Sistem Mengirimkan Notifikasi	
<b>Alur kejadian normal</b>	<b>Aktor</b>	<b>Sistem</b>
		1 Mengirimkan Notifikasi
	2. Menerima notifikasi	

### 3.1.3.4.36 Mengubah Status Perkuliahan (UC-036)

Kasus penggunaan nomor UC-036 ini adalah saat Dosen akan mengaktifkan perkuliahan atau mengakhiri perkuliahan. Spesifikasi dapat dilihat pada Tabel 3.17

**Tabel 3.17 Spesifikasi Kasus Penggunaan UC-036**

<b>Kode Use Case</b>	UC-036	
<b>Nama Use Case</b>	Mengubah Status Perkuliahan	
<b>Aktor</b>	Dosen	
<b>Deskripsi</b>	Dosen dapat mengganti status perkuliahan	
<b>Relasi</b>	-	
<b>Kondisi Awal</b>	Perkuliahan dalam keadaan belum aktif	
<b>Kondisi Akhir</b>	Perkuliahan dalam keadaan aktif	
<b>Alur kejadian normal</b>	<b>Aktor</b>	<b>Sistem</b>
	1 Memilih pilihan aktifkan kelas	
		2. Mengubah status kelas

## 3.2 Perancangan

Perancangan dalam subbab ini membahas perancangan dari aplikasi tugas akhir. Subbab ini terdiri dari lingkungan perancangan perangkat lunak, perancangan arsitektur sistem, perancangan diagram kelas, perancangan basis data, dan perancangan antarmuka pengguna.

### 3.2.1 Perancangan Arsitektur Sistem

Pada pembuatan aplikasi ini, menggunakan kerangka kerja Laravel, yang menggunakan konsep MVC (*Model, View Controller*). *Browser* digunakan untuk mengakses aplikasi Web dan *Smartphone* digunakan untuk mengakses aplikasi *mobile*. Proses bisnis yang ada pada aplikasi ditampilkan melalui antarmuka sistem.

Dalam proses integrasi antara aplikasi Web dan Mobile, digunakanlah *web-service api*. Aplikasi *mobile* akan mengirimkan *request* ke server dalam bentuk *string request*. Sistem mengirim *string request* ke *api*.. Sistem pada *web service* akan menerima data dari aplikasi dan menyimpannya ke dalam *database* yang sudah dibuat. Jika data berhasil disimpan, *query response* memberikan nilai *true* dan *web service* akan mengirimkan JSON *post response* ke aplikasi.

**Tabel 3.18 Daftar Request Web Service**

No	Request	Aktor
1	Login	Semua
2	List Kelas Semester ini	Dosen, Mahasiswa
3	List perkuliahan hari ini	Dosen, Mahasiswa
4	Unggah Wajah dan Tanda Tangan	Mahasiswa

No	Request	Aktor
5	Unduh Wajah dan Tanda Tangan	Mahasiswa
6	List Perkuliahan Minggu ini	Orang Tua
7	Aktifkan Kelas	Dosen
8	Ubah Jadwal	Dosen
9	Ubah Status Kehadiran Mahasiswa	Dosen
10	Rekap Kehadiran Kelas	Dosen

### 3.2.2 Perancangan Basis Data

Dalam membuat suatu aplikasi perangkat bergerak, diperlukan analisis kebutuhan berupa perancangan basis data. Basis data yang digunakan kelak adalah MySQL. MySQL dipilih menjadi basis data aplikasi ini karena sifat RDBMS yang *open source*, mudah digunakan, dan memiliki *performance tuning*, yaitu menangani *query* sederhana dengan cepat [1]. Sedangkan pada aplikasi *mobile*, menggunakan SQLite Database.

Rancangan basis data ditampilkan dalam bentuk *Conceptual Data Model* (selanjutnya disebut CDM) dan *Physical Data Model* (selanjutnya disebut PDM). Penjelasan lebih lengkap berupa CDM dan PDM terdapat pada Gambar 3.16 dan Gambar 3.17.

#### 3.2.2.1 Tabel Users

Tabel users adalah tabel yang digunakan untuk menyimpan data-data para pengguna yang memiliki hak akses pada sistem. Tabel ini merupakan tabel utama yang mempunyai

atribut-atribut, yaitu *email*, *password*, dan *role*. Sistem ini hanya menyediakan empat hak akses, yaitu dosen, mahasiswa, karyawan dan orang tua. Detil atribut tabel dijelaskan pada Tabel 3.19

**Tabel 3.19 Atribut Tabel Users**

Nama Kolom	Tipe Data	Keterangan
id	Int	Primary key
Nrp	Varchar(100)	Nama user
Email	Varchar(100)	Email user untuk login
Password	Varchar(100)	
Remember_token	Varchar(100)	Untuk <i>recovery</i> password
Created_at	Timestamp	Waktu data dibuat
Updated_at	Timestamp	Waktu data diubah
User_id	Int	Nip pada dosen, nrp pada mahasiswa, 1+nrp untuk orang tua
role	Int	1: Dosen, 2. TU. 3 Orang Tua 4. Mahasiswa
Token_fcm	Varchar(100)	Untuk token pengiriman <i>push notification</i>

### 3.2.2.2 Tabel Mahasiswa

Tabel mahasiswa adalah tabel yang digunakan untuk menyimpan data-data mahasiswa yaitu, NRP, email, alamat dan telepon. Detil atribut tabel dijelaskan pada Tabel 3.20

**Tabel 3.20 Atribut Tabel Mahasiswa**

Nama Kolom	Tipe Data	Keterangan
NRP	Varchar(100)	Primary key Mahasiswa
Nama	Varchar(100)	Nama Mahasiswa
Alamat	Varchar(100)	Alamat Mahasiswa
Email	Varchar(100)	Email mahasiswa
No_telpon	Varchar(100)	No Telpon Mahasiswa
Created_at	Timestamp	Waktu data dibuat
Updated_at	Timestamp	Waktu data diubah
Tanggal_lahir	Int	Tanggal lahir mahasiswa
Kode_ortu	Int	Foreign key ke tabel orang tua

### 3.2.2.3 Tabel Dosen

Tabel dosen adalah tabel yang digunakan untuk menyimpan data-data dosen yaitu, NIP, email, alamat dan telepon. Detil atribut tabel dijelaskan pada Tabel 3.21

**Tabel 3.21 Atribut Tabel Dosen**

>Nama Kolom	Tipe Data	Keterangan
NIP	Varchar(100)	Primary key Dosen
Nama	Varchar(100)	Nama Dosen
Alamat	Varchar(100)	Alamat Dosen
Email	Varchar(100)	Email Dosen
No_telpon	Varchar(100)	No Telpon Dosen
Created_at	Timestamp	Waktu data dibuat
Updated_at	Timestamp	Waktu data diubah
Tanggal_lahir	Int	Tanggal lahir Dosen
Kode_dosen	Int	Dua karakter yang merepresentasikan dosen

### 3.2.2.4 Tabel Orangtua

Tabel orangtua adalah tabel yang digunakan untuk menyimpan data-data orangtua mahasiswa yaitu, email, alamat, telepon dan nrp sebagai *foreign key* ke table mahasiswa. Detil atribut tabel dijelaskan pada Tabel 3.22

**Tabel 3.22 Atribut Tabel Orang Tua**

>Nama Kolom	Tipe Data	Keterangan
id	int	Primary key Orang tua
Nama	Varchar(100)	Nama Orang Tua
Alamat	Varchar(100)	Alamat Orang Tua
Email	Varchar(100)	Email Orang Tua
No_telpon	Varchar(100)	No Telpon Orang

		Tua
Created_at	Timestamp	Waktu data dibuat
Updated_at	Timestamp	Waktu data diubah
Nrp	Int	<i>Foreign key pada tabel mahasiswa</i>

### 3.2.2.5 Tabel Wajah

Tabel wajah adalah tempat menyimpan alamat penyimpanan foto wajah mahasiswa yang akan digunakan untuk verifikasi kehadiran. Detil atribut tabel dijelaskan pada Tabel 3.23

**Tabel 3.23 Atribut Tabel Wajah**

Nama Kolom	Tipe Data	Keterangan
id	int	Primary key wajah
foto	Varchar(100)	Menyimpan alamat penyimpanan foto wajah pada server
Nrp	Varchar(100)	<i>Foreign key ke tabel mahasiswa</i>
Created_at	Timestamp	Waktu data dibuat
Updated_at	Timestamp	Waktu data diubah

### 3.2.2.6 Tabel Signature

Tabel signature adalah tempat menyimpan alamat penyimpanan tanda tangan mahasiswa yang akan digunakan



untuk verifikasi kehadiran. Detil atribut tabel dijelaskan pada Tabel 3.24

**Tabel 3.24 Atribut Tabel Mahasiswa**

Nama Kolom	Tipe Data	Keterangan
id	int	Primary key
foto	Varchar(100)	Menyimpan alamat penyimpanan foto tanda tangan pada server
Nrp	Varchar(100)	<i>Foreign</i> key ke tabel mahasiswa
Created_at	Timestamp	Waktu data dibuat
Updated_at	Timestamp	Waktu data diubah

### 3.2.2.7 Tabel Matakuliah

Tabel yang digunakan untuk menyimpan data-data matakuliah, yakni nama, semester, rmk dan kode. Detil atribut tabel dijelaskan pada Tabel 3.25

**Tabel 3.25 Atribut Tabel Matakuliah**

Nama Kolom	Tipe Data	Keterangan
id	int	Primary key matakuliah
nama	Varchar(100)	Nama matakuliah
rmk	Varchar(100)	Ranah mata kuliah
Created_at	Timestamp	Waktu data dibuat
Updated_at	Timestamp	Waktu data

		diubah
Kode_integra	Varchar(100)	Menyimpan kode integra
Sks	Int	Jumlah Sks
Kode_matakuliah	Varchar(100)	Menyimpan singkatan matakuliah

### 3.2.2.8 Tabel Ruangan

Tabel ruangan adalah untuk menyimpan nama dan detail posisi ruangan, yaitu latitude, longitude dan latitudenya yang akan digunakan untuk pencocokan lokasi sebelum dilakukannya proses verifikasi kehadiran mahasiswa. Detil atribut tabel dijelaskan pada Tabel 3.26

**Tabel 3.26 Atribut Tabel Ruangan**

Nama Kolom	Tipe Data	Keterangan
id	int	Primary key ruangan
Kode_ruangan	Varchar(100)	
nama	Varchar(100)	Menyimpan nama ruangan
Created_at	Timestamp	Waktu data dibuat
Updated_at	Timestamp	Waktu data diubah
longitude	double	Menyimpan longitude
latitude	double	Menyimpan latitude
altitude	double	Menyimpan altitude

### 3.2.2.9 Tabel Kelas

Tabel untuk menyimpan data-data kelas. Memiliki relasi ke banyak tabel, yaitu matakuliah, ruangan, waktuperkuliahan, dan perkuliahan mahasiswa. Detil atribut tabel dijelaskan pada Tabel 3.27

**Tabel 3.27 Atribut Tabel kelas**

Nama Kolom	Tipe Data	Keterangan
id	int	Primary key kelas
Kode_kelas	Varchar(2)	A, B, C...
Kode_matakuliah	Varchar(100)	Menyimpan singkatan matakuliah
Created_at	Timestamp	Waktu data dibuat
Updated_at	Timestamp	Waktu data diubah
Kode_ruangan	Int	<i>Foreign key</i> ke tabel ruangan
Semester	Int	

### 3.2.2.10 Tabel Waktuperkuliahan

Tabel ini dibuat karena setiap kelas dimungkinkan untuk memiliki lebih dari satu jadwal, sehingga memiliki hubungan *many to many* ke tabel kelas. Detil atribut tabel dijelaskan pada Tabel 3.28

**Tabel 3.28 Atribut tabel Waktuperkuliahan**

Nama Kolom	Tipe Data	Keterangan
id	int	Primary key waktuperkuliahan
hari	Varchar(11)	Menyimpan Hari
mulai	Time	Menyimpan jam mulai
selesai	time	Menyimpan jam selesai
Created_at	Timestamp	Waktu data dibuat
Updated_at	Timestamp	Waktu data diubah

### 3.2.2.11 Tabel Perkuliahanmahasiswa

Tabel yang menyimpan detail perkuliahan, yaitu pertemuan, status perkuliahan, hari, jam mulai, jam selesai dan status dosen. Dosen dan Karyawan memiliki wewenang untuk merubah isi tabel ini. Detil atribut tabel dijelaskan pada Tabel 3.29

**Tabel 3.29 Atribut Tabel Perkuliahanmahasiswa**

Nama Kolom	Tipe Data	Keterangan
id	int	Primary key perkuliahanmahasiswa
pertemuan	int	Menyimpan Pertemuan kelas
Status_dosen	char	0 belum aktif, 1 aktif, 2 selesai
Status_perkuliahan	char	0 normal 1 pengganti
Id_kelas	int	<i>Foreign key</i> ke tabel kelas

tanggal	date	Tanggal perkuliahan diadakan
hari	Varchar(20)	Menyimpan hari
Mulai	Time	Menyimpan jam mulai
Selesai	Time	Menyimpan jam selesai
Created_at	Timestamp	Waktu data dibuat
Updated_at	Timestamp	Waktu data diubah
Id_ruangan	int	<i>Foreign key</i> ke tabel ruangan

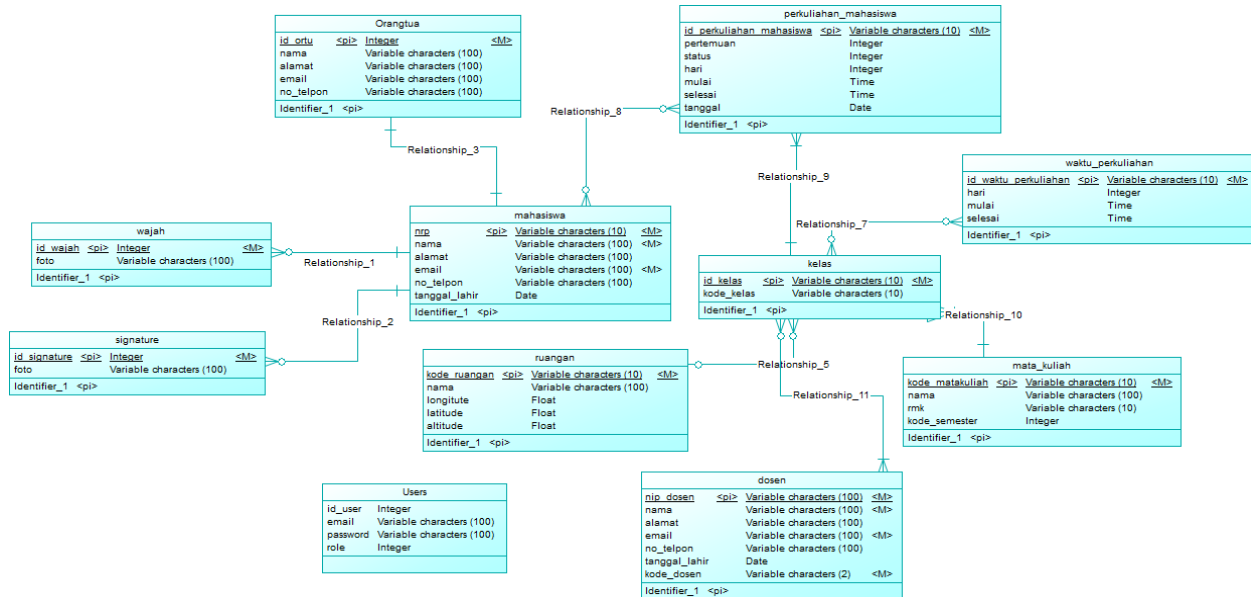
### 3.2.2.12 Tabel Kehadiran

Sebuah tabel *pivot* antara tabel mahasiswa dan perkuliahan mahasiswa, yang akan diisi oleh keterangan kehadiran mahasiswa berisi Masuk, Ijin atau Absen. Detil atribut tabel dijelaskan pada Tabel 3.30

**Tabel 3.30 Atribut Tabel Kehadiran**

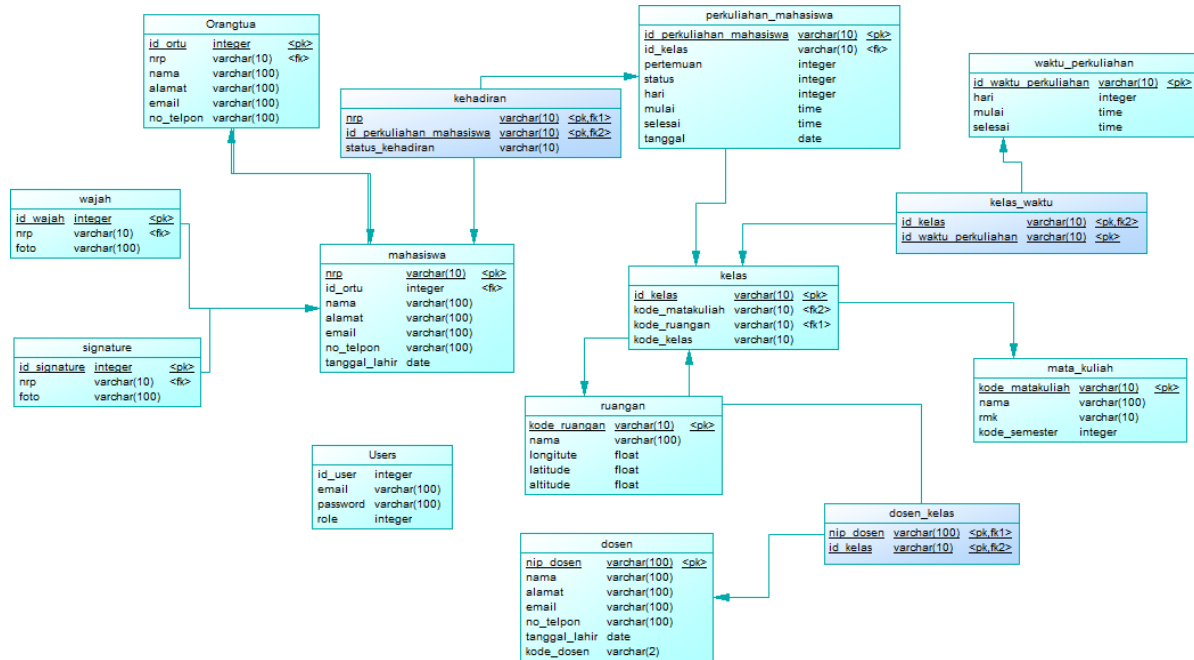
Nama Kolom	Tipe Data	Keterangan
id	int	<i>Primary key</i>
Ket_kehadiran	char	Menyimpan jam mulai
selesai	time	Menyimpan jam selesai
Created_at	Timestamp	Waktu data dibuat
Updated_at	Timestamp	Waktu data diubah

### 3.2.2.1 Conceptual Data Modeling (CDM)



Gambar 3.16 Conceptual Data Modeling

### 3.2.2.2 Physical Data Modeling (PDM)



Gambar 3.17 Physical Data Modeling

### 3.2.3 Perancangan Antarmuka Pengguna

Perancangan antarmuka pengguna merupakan hal yang penting dalam melakukan perancangan perangkat lunak. Antarmuka pengguna yang berhubungan langsung dengan aktor harus memiliki kemudahan-kemudahan dan tampilan yang menarik bagi penggunanya. Aplikasi ini memiliki empat hak akses, yaitu karyawan, dosen dan orang tua dengan halaman *login* pengguna yang sama pada aplikasi web. Sedangkan Dosen, Mahasiswa dan Orang tua pada halaman *login* pada aplikasi perangkat bergerak. Pada hak akses karyawan, aplikasi ini memiliki beberapa antarmuka, yaitu halaman kelola dosen, mahasiswa, orang tua, ruangan, matakuliah, dan kelas. Pada hak akses orang tua, terdapat halaman lihat perkuliahan anak minggu ini dan monitor kehadiran anak. Pada hak akses Dosen terdapat halaman perkuliahan hari ini, dan list kelas diampu. Pada halaman mahasiswa terdapat halaman perkuliahan hari ini dan list kelas.

#### 3.2.3.1 Rancangan Halaman Antarmuka Login Pengguna

Halaman ini digunakan oleh pengguna untuk masuk ke sistem. Pengguna harus mengisi isian *email* atau NRP atau NIP dan *password* untuk dapat masuk ke dalam sistem. Rancangan halaman dapat dilihat pada gambar 3.18





The image shows a mobile application login screen. At the top, there is a status bar with signal, Wi-Fi, and battery icons, and the time 15:45. Below the status bar, the text "Selamat Datang" (Welcome) is displayed in a bold, sans-serif font. Underneath, "Sistem Kehadiran Mahasiswa" (Student Attendance System) is written in a smaller, bold, sans-serif font. A large, stylized blue logo "IF" is centered on the screen. Below the logo, the word "LOGIN" is written in a bold, sans-serif font, flanked by two horizontal blue lines. A light blue rectangular box contains two input fields: "Username" and "Password", each with a horizontal line for text entry. Below the input fields, a blue button with the white text "masuk" (enter) is centered.

**Gambar 3.18 Rancangan Halaman Antarmuka Login Pengguna**

### **3.2.3.2 Rancangan Halaman Antarmuka Kelola Mahasiswa**

Halaman ini akan ditampilkan ketika karyawan melakukan pengelolaan data mahasiswa, meliputi tambah, ubah dan lihat. Rancangan halaman dapat dilihat pada Gambar 3.18, Gambar 3.19 dan Gambar 3.20

Daftar Mahasiswa

Tambah Mahasiswa

🔍 Cari mahasiswa

NRP	Nama	Alamat	Email	No Telpn	Tanggal Lahir	Aksi
5113100050	Bana	Perumdos	bana@gmail.com	085648281303	21-01-1995	Ubah Lihat Hapus
5113100112	Ronald	Keputih	goro@gmail.com	0856482834234	21-01-1996	Ubah Lihat Hapus

Gambar 3.19 Perancangan Antarmuka Kelola Mahasiswa

Tambah Mahasiswa

NRP

Email

Nama

Tanggal Lahir

12 May 2016

Alamat

Tambah

Gambar 3.20 Perancangan Antarmuka Tambah Mahasiswa

### Ubah Mahasiswa

NRP

Nama

Alamat

Email

Tanggal Lahir

**Gambar 3.21 Perancangan Antarmuka Ubah Mahasiswa**

Untuk perancangan antarmuka Kelola yang lain, yaitu Dosen, Orang Tua, Matakuliah, Kelas, Ruangan dan Waktu perkuliahan sama dengan perancangan antarmuka Mahasiswa. Namun data yang ditampilkan menyesuaikan dengan tabel masing-masing.

### **3.2.3.3 Rancangan Halaman Antarmuka Melihat Data Perkuliahan**

Karyawan, Dosen, Mahasiswa dan Orang Tua dapat melihat data perkuliahan. Namun batasannya berbeda, Dosen hanya dapat melihat perkuliahan yang diampunya, Mahasiswa hanya dapat melihat perkuliahan yang diampunya dan Orang Tua hanya dapat melihat perkuliahan anaknya. Rancangan halaman dapat dilihat pada Gambar 3.22, Gambar 3.23 dan Gambar 3.24

Jadwal Perkuliahan Anda Hari Ini						
Kode Matakuliah	Kode ...	Ruangan	Pertemuan ke	Jadwal	Status	Action
ERPL	A	IF-101	1	Senin 7.30-9.30	Selesai	Ubah Jadwal Peserta
BDT	A	IF-102	1	Senin 9.30-11.30	Aktif	Ubah Jadwal Peserta
MBD	A	IF-103	1	Senin 13.30-15.30	Belum Aktif	Ubah Jadwal Peserta

**Gambar 3.22 Perancangan Antarmuka Melihat Data Perkuliahan  
(Aktor: Dosen)**

Jadwal Perkuliahan Mohammad Shahbana Satriawan minggu Ini							
Kode Matakuliah	Kode ...	Ruangan	Pertemuan ke	Jadwal	Dosen	Status	Ket Kehadiran
ERPL	A	IF-101	1	Senin 7.30-9.30	Dwi Sunaryono	Selesai	Hadir
BDT	A	IF-102	1	Selasa 9.30-11.30	Sanwosri	Aktif	Ijin
MBD	A	IF-103	1	Rabu 13.30-15.30	Dwi Sunaryono	Belum Aktif	

**Gambar 3.23 Perancangan Antarmuka Melihat Data Perkuliahan  
(Aktor: Orang Tua)**



**Gambar 3.24 Perancangan Antarmuka Melihat Data Perkuliahan  
(Aktor: Mahasiswa dan Orang Tua)**

#### **3.2.3.4 Rancangan Halaman Antarmuka Mengubah Jadwal Perkuliahan**

Halaman ini akan ditampilkan saat dosen atau karyawan hendak mengganti jadwal suatu perkuliahan. Rancangan halaman dapat dilihat pada Gambar 3.25 dan Gambar 3.26

SIKEMAS

### Penjadwalan Ulang

Anda dapat melakukan penjadwalan ulang mata kuliah yang Anda ampu

KI141425 - Pemrograman Perangkat Bergerak - A

**KI141425**  
**Pemrograman Perangkat Bergerak**  
 Kelas : A Ruang : IF - 105a  
 Hari : Kamis  
 Waktu : 12.30 - 15.00

**Ganti Jadwal**

**Gambar 3.25 Perancangan Antarmuka Mengubah Jadwal Perkuliahan (Mobile)**

### Ubah Jadwal

Kelas	Tanggal
<input type="text" value="Sistem Basis Data"/>	<input type="text" value="12 May 2016"/>
Kode	Waktu
<input type="text" value="A"/>	<input type="text" value="Select"/>
Pertemuan	Ruangan
<input type="text" value="2"/>	<input type="text" value="Select"/>

**Gambar 3.26 Perancangan Antarmuka Mengubah Jadwal Perkuliahan (Web)**

### 3.2.3.5 Rancangan Halaman Antarmuka Melihat Data Kehadiran Mahasiswa

Halaman ini akan ditampilkan saat dosen atau karyawan hendak melihat rekap kehadiran mahasiswa pada suatu kelas. Rancangan halaman dapat dilihat pada Gambar 3.27 dan Gambar 3.28

Kelas ERPL A																
No	NRP	Nama	Pertemuan													
			1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	5111100138	Sandri Ayunier K	M	M	I	A	M	I	M	M	M	M	M	M	M	A
2	5111100198	Hery Nugroho	M	M	M	M	M	I	M	A	M	M	M	M	M	M

**Gambar 3.27 Rancangan Halaman Antarmuka Melihat Data Kehadiran Mahasiswa (Web)**



**Gambar 3.28 Rancangan Halaman Antarmuka Melihat Data Kehadiran Mahasiswa (Mobile)**



### 3.2.3.6 Rancangan Halaman Antarmuka Menambah Peserta Kelas

Halaman ini akan ditampilkan saat karyawan akan menambahkan mahasiswa ke dalam kelas. Rancangan halaman dapat dilihat pada Tabel 3.29



The image shows a web interface for adding class participants. It has a title 'Tambah Peserta Kelas' at the top. Below the title, there are three main sections. The first section is labeled 'Format Upload' and contains a button 'Unduh Format'. The second section is labeled 'File' and contains a button 'Pilih File' and a text label 'Peserta ERPL.xls'. The third section contains a button 'Unggah'.

Tambah Peserta Kelas	
Format Upload	Unduh Format
File	Pilih File      Peserta ERPL.xls
Unggah	

**Gambar 3.29 Rancangan Halaman Antarmuka Menambah Peserta Kelas**

*[Halaman ini sengaja dikosongkan]*

## **BAB IV IMPLEMENTASI**

Bab ini membahas implementasi yang dilakukan berdasarkan rancangan yang telah dijabarkan pada bab sebelumnya. Sebelum penjelasan implementasi akan ditunjukkan terlebih dahulu lingkungan untuk melakukan implementasi.

Pada bagian implementasi ini juga akan dijelaskan mengenai fungsi-fungsi yang digunakan dalam program tugas akhir ini dan disertai dengan kode sumber masing-masing fungsi utama.

### **4.1 Lingkungan Implementasi Perangkat Lunak**

Spesifikasi perangkat keras serta perangkat lunak yang digunakan dalam tahap implementasi perangkat lunak tugas akhir ini seperti dijelaskan pada

Tabel 4.1.

**Tabel 4.1 Lingkungan Implementasi Perangkat Lunak**

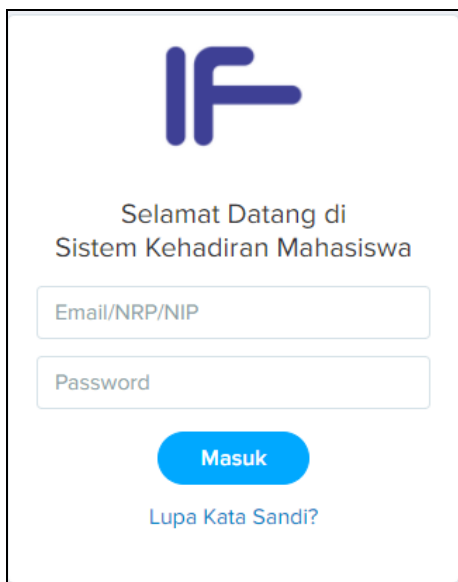
Perangkat Keras	Komputer	Acer Aspire
	Prosesor	Intel® Core™ i3-2120M CPU @ 3.3GHz
	Memori Primer	8 GB
	Memori Sekunder	500 GB
Perangkat Lunak	Sistem Operasi	Windows 8.1 Pro 64-bit

	Perangkat Lunak	Android Studio v2.2.2, Paint, Sybase PowerDesigner 16.5, Microsoft Word 2013
--	-----------------	---

## 4.2 Implementasi Antarmuka Pengguna

Implementasi antarmuka pengguna berbasis perangkat bergerak ini menggunakan berkas XML yang dibangun pada lingkungan pengembangan Android Studio. Pada subbab ini akan dijelaskan dan ditampilkan tampilan halaman XML sesuai dengan rancangan antarmuka yang terdapat pada bab III.

### 4.2.1 Implementasi Halaman Antarmuka Login



**Gambar 4.1 Implementasi Halaman Antarmuka Login Aplikasi Web**

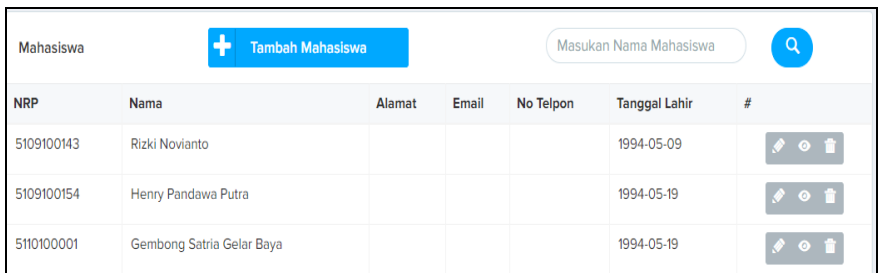


**Gambar 4.2 Implementasi Halaman Antarmuka Login Aplikasi Mobile**

Halaman antarmuka login pada Gambar 4.1 dan Gambar 4.2 merupakan halaman yang digunakan oleh pengguna untuk masuk ke sistem dan mengakses halaman utama. Terdapat tempat isian untuk *email* atau NRP atau NIP dan *password*, serta tombol *login*.

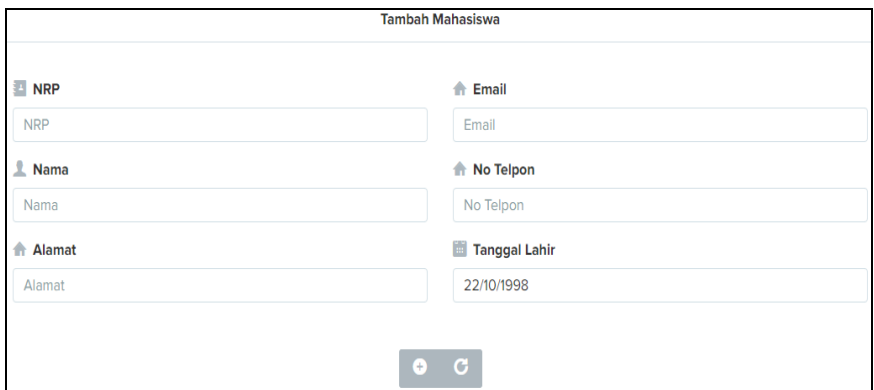
#### 4.2.2 Implementasi Halaman Antarmuka Kelola Mahasiswa

Halaman antarmuka kelola mahasiswa ditampilkan ketika karyawan melakukan pengelolaan data mahasiswa, meliputi tambah, ubah, lihat dan hapus. Implementasi dapat dilihat pada Gambar 4.3 dan Gambar 4.4



NRP	Nama	Alamat	Email	No Telpn	Tanggal Lahir	#
5109100143	Rizki Novianto				1994-05-09	
5109100154	Henry Pandawa Putra				1994-05-19	
5110100001	Gembong Satria Gelar Baya				1994-05-19	

**Gambar 4.3 Implementasi Halaman Antarmuka Kelola Mahasiswa**



**Tambah Mahasiswa**

**NRP**

**Email**

**Nama**

**No Telpn**

**Alamat**

**Tanggal Lahir**














**Gambar 4.4 Implementasi Halaman Antarmuka Tambah Mahasiswa**

Untuk implementasi antarmuka Kelola yang lain, yaitu Dosen, Orang Tua, Matakuliah, Ruangan dan

Waktuperkuliahan sama dengan implementasi antarmuka Mahasiswa. Namun data yang ditampilkan menyesuaikan dengan tabel masing-masing.

### 4.2.3 Implementasi Halaman Antarmuka Kelola Kelas

Halaman antarmuka kelola kelas ditampilkan ketika karyawan melakukan pengelolaan data kelas, meliputi tambah, ubah, lihat dan hapus. Implementasi dapat dilihat pada Gambar 4.3 dan Gambar 4.4

Kelas								
		 Tambah Kelas		<input type="text" value="Masukan Nama Kelas"/> 				
Kode Mata Kuliah	Kode Kelas	Kode Ruang	Jadwal	Dosen Pengajar	Perkuliahan	Input Peserta	Rekap Absensi	Action
ERPL	A	IF-101	Selasa,13:30:00-16:30:00	Dwi Sunaryono Sarwosri			 	 
BDT	A	IF-106	Kamis,13:30:00-16:30:00	Dwi Sunaryono Sarwosri			 	 
MBD	D	IF-105b	Rabu,11:30:00-13:30:00	Sarwosri			 	 

**Gambar 4.5 Implementasi Halaman Antarmuka Kelola Kelas**

### Gambar 4.6 Implementasi Halaman Antarmuka Menambah



Perkuliahan Kelas Basis Data Terdistribusi A						
Pertemuan	Ruangan	Tanggal	Jadwal	Status Kelas	Status Jadwal	Rekap Kehadiran
1	IF-106	2017-02-09	Kamis,13:30:00-16:30:00	Selesai	Normal	
2	IF-106	2017-02-16	Kamis,13:30:00-16:30:00	Selesai	Pengganti	
3	IF-106	2017-02-23	Kamis,13:30:00-16:30:00	Aktif	Normal	

**Gambar 4.7 Implementasi Halaman Antarmuka Melihat Data Perkuliahan (Web)**



**Gambar 4.8 Implementasi Halaman Antarmuka Melihat Data Perkuliahan (Mobile)**



**Gambar 4.9 Implementasi Mahasiswa Melihat Perkuliahan Hari Ini**

Jadwal Kuliah Mohammad Shahbana Satriawan Minggu Ini							
Mata Kuliah	Kelas	Pertemuan	Tanggal	Jadwal	Dosen	Status Perkuliahan	Ket Kehadiran
Ekonomi Rekayasa Perangkat Lunak	A	16	2017-06-04	Jumat, 14:30:00-17:00:00	Dwi Sunaryono Sarwosri	Selesai	M
Basis Data Terdistribusi	A	16	2017-06-04	Jumat, 23:02:00-23:59:00	Dwi Sunaryono Sarwosri	Aktif	M

**Gambar 4.10 Implementasi Orang Tua Melihat Perkuliahan anak**

#### 4.2.5 Implementasi Halaman Antarmuka Mengubah Jadwal Perkuliahan

Halaman ini akan ditampilkan saat dosen atau karyawan hendak mengganti jadwal suatu perkuliahan. Implementasi rancangan halaman dapat dilihat pada Gambar 4.12 dan Gambar 4.13

The screenshot shows a web form titled "Ganti Jadwal". It contains several input fields and dropdown menus arranged in two columns. The first column has three fields: "Nama" (with a calendar icon) containing "Ekonomi Rekayasa Perangkat Lunak", "Kode" (with a person icon) containing "A", and "Pertemuan" (with a person icon) containing "16". The second column has three fields: "Pilih tanggal pengganti" (with a calendar icon) containing "09/06/2017", "Waktu" (with a clock icon) containing "11:30:00", and "Ruangan" (with a building icon) containing "IF-105b". At the bottom center, there are two buttons: a plus sign (+) and a refresh/circular arrow icon.

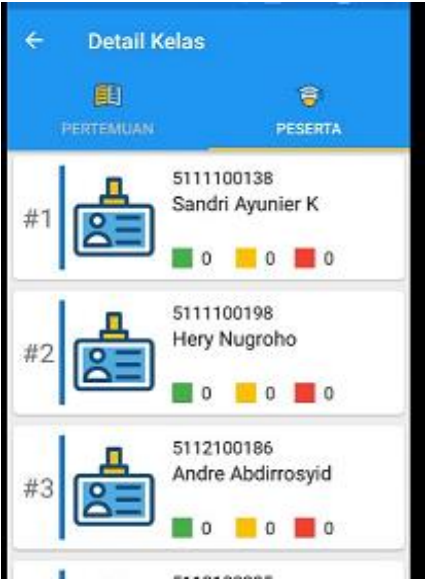
**Gambar 4.11 Implementasi Halaman Mengubah Jadwal Perkuliahan (Web)**

#### 4.2.6 Implementasi Halaman Melihat Data Kehadiran Mahasiswa

Halaman ini akan ditampilkan saat dosen atau karyawan hendak melihat rekap kehadiran mahasiswa pada suatu kelas. Implementasi halaman dapat dilihat pada Gambar 4.12 dan Gambar 4.13

Kelas ERPL A																	
No	NRP	Nama	Pertemuan														
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	5111100138	Sandri Ayunier K	M	M	I	A	M	I	M	M	M	M	M	M	M	M	A
2	5111100198	Hery Nugroho	M	M	M	M	M	I	M	A	M	M	M	M	M	M	M
3	5112100186	Andre Abdirrosyid	M	M	M	M	A	I	M	M	M	M	I	M	M	M	M

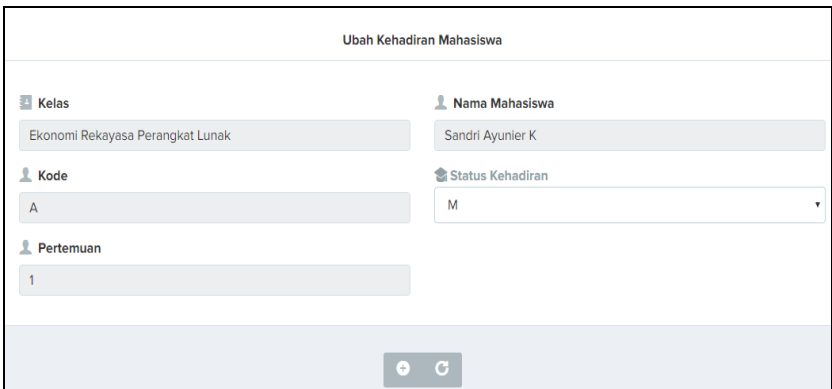
Gambar 4.12 Implementasi Halaman Melihat Data Kehadiran Mahasiswa (Web)



Gambar 4.13 Implementasi Halaman Melihat Data Kehadiran Mahasiswa (Mobile)

#### 4.2.7 Implementasi Halaman Antarmuka Mengubah Status Kehadiran Mahasiswa

Halaman ini akan ditampilkan saat dosen atau karyawan hendak mengganti status kehadiran mahasiswa pada suatu kelas. Implementasi halaman dapat dilihat pada Gambar 4.14 dan Gambar 4.15

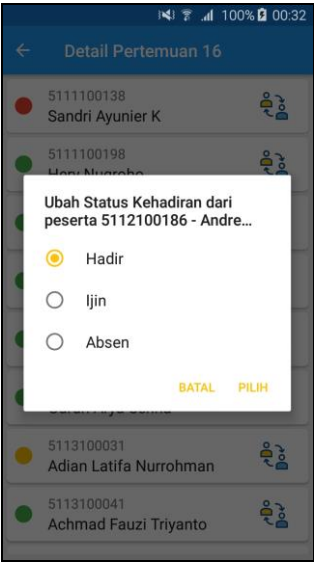


The screenshot shows a web form titled "Ubah Kehadiran Mahasiswa". The form contains several input fields and a dropdown menu:

- Kelas:** A text input field containing "Ekonomi Rekayasa Perangkat Lunak".
- Nama Mahasiswa:** A text input field containing "Sandri Ayunier K".
- Kode:** A text input field containing "A".
- Status Kehadiran:** A dropdown menu with "M" selected.
- Pertemuan:** A text input field containing "1".

At the bottom right of the form, there are two buttons: a plus sign (+) and a refresh/cancel icon (G).

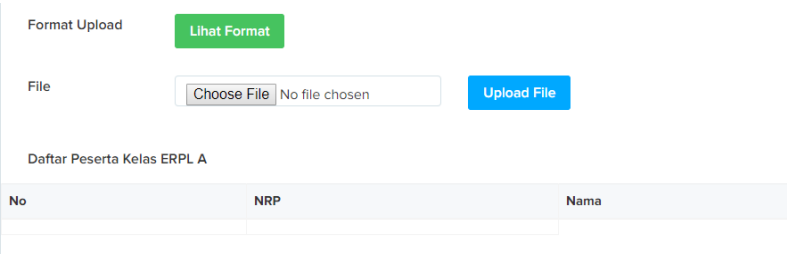
**Gambar 4.14 Implementasi Halaman Ubah Status Kehadiran Mahasiswa (Web)**



**Gambar 4.15 Implementasi Halaman Ubah Status Kehadiran Mahasiswa (Mobile)**

**4.2.8 Implementasi Halaman Antarmuka Menambah Peserta Kelas**

Halaman ini akan ditampilkan saat karyawan hendak menambahkan peserta kelas ke suatu kelas. Implementasi halaman dapat dilihat pada Gambar 4.16



**Gambar 4.16 Implementasi Halaman Antarmuka Menambah peserta Kelas**

### 4.3 Implementasi Kasus Penggunaan

Implementasi kasus penggunaan aplikasi ini menggunakan kerangka kerja Laravel pada aplikasi web dan menggunakan bahasa pemrograman Java pada aplikasi perangkat bergerak yang dibangun pada lingkungan pengembangan Android Studio. Di sisi pengembangan *web service*, bahasa yang digunakan adalah PHP yang dibangun dengan kerangka kerja Laravel. Pada subbab ini akan menjelaskan dan menampilkan kode sumber pada beberapa kasus penggunaan.

#### 4.3.1 Implementasi Kasus Penggunaan Menambah Mahasiswa

Pada kasus penggunaan ini, digunakan untuk menambahkan mahasiswa ke dalam sistem. Kode Sumber dan penjelasannya dapat dilihat pada Kode Sumber 4.1 dan Tabel 4.2

```

1.  public function tambah_mahasiswa()
2.  {
3.      if (Request::isMethod('get')) {
4.          return View::make('tu.mahasiswa.tambahmahasiswa');
5.      } else if (Request::isMethod('post')) {
6.          $data = Input::all();
7.          $tanggal = substr($data['tanggal_lahir'],0,2);
8.          $bulan = substr($data['tanggal_lahir'],3,2);
9.          $tahun = substr($data['tanggal_lahir'],6,4);
10.
11.          $gabung= $tahun."-".$bulan."-".$tanggal;
12.
13.          Mahasiswa::insertGetId(array(
14.              'nama' => $data['nama'],
15.              'nrp' => $data['nrp'],
16.              'alamat' => $data['alamat'],

```

```
17.         'email' => $data['email'],
18.         'no_telpon' => $data['no_telpon']
19.     ],
20.         'tanggal_lahir' => $gabung,
21.     ));
22.     return redirect('/kelola_mahasiswa')
23. ;
24. }
```

Kode Sumber 4.1 Menambahkan Data Mahasiswa

Tabel 4.2 Penjelasan Kode Sumber 4-1

No. Baris	Kegunaan
6	Mengambil data input
7-11	Mengubah data tanggal menjadi format yang diinginkan oleh tipe data date.
13-20	Insert data ke table mahasiswa menggunakan fitur eloquent laravel

4.3.2 Implementasi Kasus Penggunaan Mengubah Data Mahasiswa

Pada kasus penggunaan ini, digunakan untuk mengubah data mahasiswa yang telah ada dalam sistem. Kode Sumber dan penjelasannya dapat dilihat pada Kode Sumber 4.2 dan Tabel 4.2

```
1. public function update_mahasiswa($id)
2. {
3.     if (Request::isMethod('get')) {
4.         $this->data = array();
5.         $this->data = Mahasiswa::find($id);
6.         $tanggal = substr($this->data->tanggal_lahir,8,2);
7.         $bulan = substr($this->data->
```



```

>tanggal_lahir,5,2);
8.         $tahun = substr($this->data-
>tanggal_lahir,0,4);
9.         $gabung= $tanggal."/".$bulan."/".$ta
hun;
10.         return view('tu.mahasiswa.updatemaha
siswa',array('mahasiswa' => $this-
>data, 'tanggal' => $gabung));
11.     } else if (Request::isMethod('post')) {

12.         $data = Input::all();
13.         $tanggal = substr($data['tanggal_lahir'],0,2);

14.         $bulan = substr($data['tanggal_lahir'],3,2);

15.         $tahun = substr($data['tanggal_lahir'],6,4);

16.         $gabung= $tahun."-".$bulan."-".$tanggal;
17.         Mahasiswa::where('id',$id)-
>update(array(
18.             'nama' => $data['nama'],
19.             'nrp' => $data['nrp'],
20.             'alamat' => $data['alamat'],
21.             'email' => $data['email'],
22.             'no_telpon' => $data['no_telpon']
23.         ],
24.             'tanggal_lahir' => $gabung,
25.         ));
26.         return redirect('/kelola_mahasiswa')
;
27.     }
28. }
29. }

```

**Kode Sumber 4.2 Mengubah Data Mahasiswa**

**Tabel 4.3 Penjelasan Kode Sumber 4-2**

No. Baris	Kegunaan
5	Mengambil data input
6-11	Mengubah data tanggal menjadi format yang diinginkan oleh tipe data date.
13-20	<i>Update</i> data ke tabel mahasiswa menggunakan fitur <i>eloquent</i> laravel

#### 4.3.3 Implementasi Kasus Penggunaan Menghapus Data Mahasiswa

Pada kasus penggunaan ini, digunakan untuk menghapus data mahasiswa yang telah ada dalam sistem. Kode Sumber dan penjelasannya dapat dilihat pada Kode Sumber 4.3 dan Tabel 4.3

```

1. public function delete($id)
2. {
3.     $item = Mahasiswa::find($id);
4.     $item->delete();
5.     return back();
6. }

```

**Kode Sumber 4.3 Menghapus Data Mahasiswa****Tabel 4.4 Penjelasan Kode Sumber 4-2**

No. Baris	Kegunaan
3	Mengambil data mahasiswa
4	Menghapus data mahasiswa

### 4.3.4 Implementasi Menambah Data Kelas

Pada kasus penggunaan ini, digunakan untuk menambahkan kelas. Agar jadwal, kode kelas dan ruangan tidak bertabrakan, maka diimplementasikan *Javascript* untuk memastikan kode, jadwal dan ruangan yang tersedia sesuai dengan yang ada di database. Kode Sumber dan penjelasannya dapat dilihat pada Kode Sumber 4.4, 4.5 dan 4.6

```

1. <script type="text/javascript">
2.     $(document).ready(function() {
3.         $('select[name="matakuliah"]').on('change',
4.             function() {
5.                 var stateID = $(this).val();
6.                 if(stateID) {
7.                     $.ajax({
8.                         url: '/sikemas/public/select_kode_kelas/'+stateID
9.                         ,
10.                        type: "GET",
11.                        dataType: "json",
12.                        success: function(data) {
13.                            $('select[name="kode"]').empty();
14.                            $.each(data, function(id, nama) {
15.                                $('select[name="kode"]').append('<option
16.                                    value="'+ id +'>' + id + '</option>');
17.                            });
18.                        }
19.                    });
20.                }else{
21.                    $('select[name="kode"]').empty();
22.                }
23.            });
24.     });
25. </script>

```

```

26. <script type="text/javascript">
27.     $(document).ready(function() {
28.     $('select[name="hari1"]').on('change', function
    () {
29.         var stateID = $(this).val();
30.         if(stateID) {
31.             $.ajax({
32.                 url: '/sikemas/public/select_waktu1/'+stateID,
33.                 type: "GET",
34.                 dataType: "json",
35.                 success: function(data) {
36.
37.                     $('select[name="waktu1"]').empty();
38.                     $('select[name="waktu1"]').append('
<option value="">Pilih</option>');
39.                     $.each(data, function(id_mulaiselesai, mulai)
    {
40.
41.                         $('select[name="waktu1"]').append('<option va
    lue="'+ id_mulaiselesai +' ">'+ mulai+'</option>')
    );
42.
43.
44.                     }
45.                 });
46.             }else{
47.                 $('select[name="waktu1"]').append('<option valu
    e="">Jadwal Kosong</option>');
48.
49.             }
50.         });
51.     }</script>
52.

```

```

53. <script type="text/javascript">
54.     $(document).ready(function() {
55.         $('select[name="waktu1"]').on('change',
56.         function() {
57.             var stateID = $(this).val();
58.             if(stateID) {
59.                 $.ajax({
60.                     type: "GET",
61.                     dataType: "json",
62.                     success: function(data) {
63.
64.                         $('select[name="ruangan"]').empty();
65.                         $('select[name="ruangan"]').append('
66.                         <option value="">Pilih</option>');
67.                         $.each(data, function(nama,id) {
68.                             $('select[name="ruangan"]').append
69.                             ('<option value="'+ id +' ">'+ nama+'</option>');
70.
71.                         });
72.                     });
73.                 }else{
74.                     $('select[name="ruangan"]').append
75.                     ('<option value="">Ruangan Penuh</option>');
76.                 }
77.             });
78. </script>

```

**Kode Sumber 4.4 Javascript untuk mengatur agar tidak ada kode, jadwal dan ruangan yang bertabrakan**

**Tabel 4.5 Penjelasan Kode Sumber 4.4**

No. Baris	Kegunaan
2-7	Jika terjadi perubahan pada dropdown matakuliah, maka akan memanggil url ../select_kode_kelas/{id_matakuliah}
10-23	Mengembalikan kode kelas yang tersedia dan mengisi pilihan pada dropdown kode_kelas
27-32	Jika terjadi perubahan pada dropdown hari, maka akan memanggil url ../select_waktu1/{id_hari}
37-50	Mengembalikan waktu yang tersedia dan mengisi pilihan pada dropdown waktu
54-59	Jika terjadi perubahan pada dropdown waktu, maka akan memanggil url ../select_ruangan/{id_waktu}
64-77	Mengembalikan ruangan yang tersedia dan mengisi pilihan pada dropdown ruangan

```

1. public function select_kode_kelas($kode_matkul)
2.     {
3.
4.         $matkul=Matakuliah::where('kode_matakuliah',
           $kode_matkul)->first();
5.         $exclude= Kelas::where('kode_matakuliah',$kod
           e_matkul)->select('kelas.kode_kelas')->get()-
           >toArray();
6.         $sub =DB::table('kode_kelas')-
           >whereNotIn('nama',$exclude)-
           >pluck('id','nama');
7.         $sub->semester=$matkul->semester;
8.         return json_encode($sub);
9.     }
10.
11.

```

```

12.     public function select_waktu1($id_hari)
13.     {
14.         $sub=Waktuperkuliahan::where('id_hari',$i
d_hari)->pluck('mulai','id');
15.         return json_encode($sub);
16.     }
17.
18.
19.     public function select_ruangan($id_waktu)
20.     {
21.         $exclude=Kelas::whereHas('kelaswaktu','fu
nction($q) use($id_waktu){
22. $q->where('id_waktuperkuliahan',$id_waktu);
23.     }->select('kelas.kode_ruangan')->get()-
>toArray();
24.         $sub=Ruangan::whereNotIn('id',$exclude)-
>pluck('id','nama');
25.         return json_encode($sub);
26.     }

```

**Kode Sumber 4.5 Fungsi pada KelasController**

**Tabel 4.6 Penjelasan Kode Sumber 4.5**

No. Baris	Kegunaan
4	Mengambil data matakuliah sesuai input
5	Mengambil kode kelas yang telah dipakai
6	Menyeleksi kode kelas yang tersedia dengan cara membuang kode kelas yang telah dipakai
8	Mengembalikan nilai kode kelas yang telah terseleksi
12- 15	Mengembalikan nilai waktu yang tersedia sesuai input
21- 23	Mengambil id_ruangan yang telah dipakai pada waktu tersebut

24	Menyeleksi ruangan yang tersedia dengan cara membuang ruangan yang telah dipakai
25	Mengembalikan nilai ruangan yang telah terseleksi

```

1. public function tambah_kelas()
2.
3.     if (Request::isMethod('post')) {
4.         $data = Input::all();
5.         $carinama= DB::table('matakuliah')-
>where('kode_matakuliah', '=', $data['matakuliah'])-
>first();
6.         $cariruang= DB::table('ruangan')-
>where('id', '=', $data['ruangan'])->first();
7.         $makuliah=Matakuliah::where('kode_maku
liah', $data['matakuliah'])-
>select('matakuliah.kode_semester')-
>first();
8.         Kelas::insertGetId(array(
9.             'kode_matakuliah' => $data['matakuliah'],
10.            'kode_kelas' => $data['kode'],
11.            'kode_ruangan' => $data['ruangan'],
12.
13.            'nama_kelas' => $carinama->nama,
14.            'nama_ruangan' => $cariruang->nama,
15.            'kode_semester'=>$makuliah-
>kode_semester
16.        ));
17.        $idkelas= DB::table('kelas')-
>max('id');
18.        $caridosen= DB::table('dosen')-
>where('id', '=', $data['dosen1'])->first();
19.        Dosenkelas::insertGetId(array(
20.            'kode_dosen'=>$data['dosen1'],
21.            'id_kelas'=>$idkelas,
22.            'nama_dosen'=>$caridosen->nama
23.        ));
24.        if($data['dosen2']!=null){
25.            $caridosen= DB::table('dosen')-
>where('id', '=', $data['dosen2'])->first();
26.            Dosenkelas::insertGetId(array(

```



```

26.         'kode_dosen'=>$data['dosen2'],
27.         'id_kelas'=>$idkelas,
28.         'nama_dosen'=>$caridosen->nama
29.     ));
30.     }
31.     $cariid= DB::table('waktuperkuliahahan')-
>where('id','=',$data['waktu1'])->first();
32.     Kelaswaktu::insertGetId(array(
33.         'id_waktuperkuliahahan'=>$data['wa
ktu1'],
34.         'id_kelas'=>$idkelas
35.     ));
36.     $dt = Carbon::create(2017, 2, 5);
37.     $dt=$dt->addDay($data['hari1']);
38.     for ($x = 1; $x<17; $x++) {
39.         $display=$dt->toDateString();
40.         Perkuliahanmahasiswa::insertGetId(ar
ray(
41.             'pertemuan' => $x,
42.             'status_perkuliahahan'=>0,
43.             'status_dosen'=>0,
44.             'id_kelas'=>$idkelas,
45.             'hari'=>$cariid->hari,
46.             'mulai'=>$cariid->mulai,
47.             'selesai'=>$cariid->selesai,
48.             'tanggal'=>$display,
49.             'id_ruangan'=>$data['ruangan']
50.         ));
51.         $dt=$dt->addDay(7);
52.     }
53.     return redirect('/kelola_kelas');
54. }
55. }

```

**Kode Sumber 4.6 Fungsi tambah\_kelas**

**Tabel 4.6 Penjelasan Kode Sumber 4-4**

No. Baris	Kegunaan
4	Mengambil Data Input
8	Melakukan <i>insert</i> ke tabel kelas melalui <i>eloquent</i>
18-22	Melakukan <i>insert</i> ke tabel dosenkelas melalui <i>eloquent</i> . Berisi baris id_kelas dan dosen yang telah dipilih
23-30	Melakukan <i>insert</i> ke tabel dosenkelas jika dosen ke dua diisi.
36-37	Membuat format tanggal pertemuan pertama perkuliahan dengan <i>helper</i> Carbon
40-50	Melakukan <i>insert</i> ke tabel perkuliahanmahasiswa.
51	Menambahkan 7 hari pada variabel dt untuk membuat tanggal perkuliahan berikutnya

#### 4.3.5 Implementasi Kegiatan Validasi Kehadiran

Pada implementasi ini, diintegrasikan ketiga Tugas Akhir lain, yaitu pencocokan lokasi, pencocokan wajah dan pencocokan tanda tangan. Dimulai saat perkuliahan telah aktif, Mahasiswa mulai mencocokkan lokasinya dengan ruang kelas, seperti pada Gambar 4.17. Jika telah cocok barulah dapat melakukan kegiatan validasi kehadiran melalui pencocokan wajah atau pencocokan tanda tangan.

**Gambar 4.17 Pencocokan lokasi**

```

1. searchLocationButton.setOnClickListener(new View
   .OnClickListener() {
2.         @Override
3.         public void onClick(View arg0) {
4.             getContext().startService(intent
5.             );
6.             showSearchingLocation();
7.         }
8.     });

```

**Kode Sumber 4.7 OnClickListener Tombol Pencocokan Lokasi**

**Tabel 4.7 Penjelasan Kode Sumber 4.7**

No. Baris	Kegunaan
1	Inisiasi OnClickListener pada searchLocationButton
3-5	Menjalankan Service pencocokan lokasi



**Gambar 4.18 Mencocokkan lokasi dengan gambar kelas**

```

1. public void onClick(int buttonId, String idPerku
   liahan, String kodeRuangan) {

```



```

        tau " +
26.     "lakukan pencarian lokasi ulang", Toast.LENGTH_S
        HORT).show();
27.     }
28.     break;
29. }
30. }

```

**Kode Sumber 4.8 Pencocokan lokasi dengan ruang kelas**

**Tabel 4.8 Penjelasan Kode Sumber 4.8**

No. Baris	Kegunaan
3	Saat akan masuk ke halaman validasi tanda tangan
4-9	Jika lokasi cocok dengan ruangan
10-15	Jika lokasi tidak cocok, maka akan mengeluarkan alert "Pastikan Anda berada pada ruangan yang benar atau "
17	Saat akan masuk ke halaman validasi tanda tangan
18-23	Jika lokasi cocok dengan ruangan
24-28	Jika lokasi tidak cocok, maka akan mengeluarkan alert "Pastikan Anda berada pada ruangan yang benar atau "

#### 4.3.6 Implementasi Push Notification

Pada Aplikasi Mobile ini, diimplementasikan *push notification* kepada Aktor Orang Tua, Dosen dan Mahasiswa. melalui Firebase Cloud Messaging. Di saat pergantian jadwal perkuliahan, Mahasiswa akan mendapat notifikasi. Lalu saat mahasiswa absen atau ijin pada perkuliahan, notifikasi akan dikirimkan kepada orang tua mahasiswa tersebut. Penjelasan implementasi notifikasi saat mahasiswa ijin pada perkuliahan dijelaskan pada

```

1. if( $request->input('ket_kehadiran')=="I"){
2.
3.     $optionBuilder = new OptionsBuilder();
4.     $optionBuilder->setTimeToLive(60*20);
5.
6.     $notificationBuilder = new PayloadNotification
       Builder('');
7.     $khd=Kehadiran::where('id_perkuliahan
       mahasiswa',$request->input('id_perkuliahan'))-
       >where('id_mahasiswa',$idmhs->id)->first();
8.     $abc=$khd-
       >perkuliahanmahasiswa->kelas;
9.
10.    $quer="1".$khd->mahasiswa->nrp;
11.
12.    $notificationBuilder-
       >setBody('Anak Anda Izin pada perkuliahan '.$abc
       >kode_matakuliah." karena ".$request-
       >input('pesan'))
13.    >setSound('default');
14.    $dataBuilder = new PayloadDa
       taBuilder();
15.    $dataBuilder-
       >addData(['a_data' => 'my_data']);
16.
17.    $option = $optionBuilder-
       >build();
18.    $notification = $notificati
       onBuilder->build();
19.    $data = $dataBuilder-
       >build();
20.    $token= User::where('user_id
       ', $quer)->select('token_fcm')->first()-
       >toArray();
21.
22.
23.
24. $downstreamResponse = FCM::sendTo($token, $opti
       on, $notification, $data);

```

25.

**Kode Sumber 4.9 Pengiriman Notifikasi ke Aplikasi Mobile****Tabel 4.9 Penjelasan Kode Sumber 4.9**

<b>No. Baris</b>	<b>Kegunaan</b>
1	Saat status kehadiran bernilai “T” atau Ijin
3-4	Mengatur “Option” notifikasi
7-8	Mengambil data kelas
12	Menyusun isi dari notifikasi
14-19	Build parameter yang dibutuhkan oleh firebase cloud messaging
20	Mencari data token milik orang tua, yang merupakan alamat <i>device</i> agar data tepat dikirim
24	Mengirimkan Notifikasi

*[Halaman ini sengaja dikosongkan]*



## BAB V

### PENGUJIAN DAN EVALUASI

Bab ini membahas tentang pengujian dan evaluasi pada perangkat lunak yang dibangun untuk tugas akhir ini. Pengujian dilakukan pada kasus penggunaan dari sistem perangkat lunak.

#### 5.1 Lingkungan Pengujian

Pada proses pengujian perangkat lunak, dibutuhkan suatu lingkungan pengujian yang sesuai dengan standar kebutuhan. Lingkungan pengujian dalam tugas akhir ini dilakukan pada setiap kasus penggunaan. Spesifikasi masing-masing lingkungan pengujian dijabarkan pada Tabel 5.1. dan Tabel 5.2

**Tabel 5.1 Lingkungan Pengujian Fungsionalitas Aplikasi Mobile**

Spesifikasi	Deskripsi
Jenis Perangkat	<i>Smartphone</i>
Merek Perangkat	Xiaomi Redmi 3 Pro
Sistem Operasi	Android Lollipop
Memori Internal	16 GB
RAM	3 GB

**Tabel 5.2 Lingkungan Pengujian Fungsionalitas Aplikasi Web**

Perangkat Keras	Komputer	Acer Aspire
	Prosesor	Intel® Core™ i3-2120M CPU @ 3.3GHz
	Memori Primer	8 GB
	Memori Sekunder	500 GB
Perangkat Lunak	Sistem Operasi	Windows 8.1 Pro 64-bit
	Perangkat Lunak	Browser Mozilla Firefox

## 5.2 Pengujian Fungsionalitas

Pengujian fungsionalitas ini adalah pengujian fungsi-fungsi yang berjalan pada aplikasi berdasarkan kasus penggunaan. Pengujian ini dibagi menjadi empat aktor, yaitu Karyawan, Dosen, Mahasiswa dan Orang Tua

### 5.2.1 Pengujian Fungsionalitas Pada Aktor Karyawan

Pengujian Fungsionalitas pada aktor karyawan, dilakukan oleh Pak Yudi Mulyono selaku karyawan Teknik Informatika ITS.

### 5.2.1.1 Pengujian Fungsionalitas Menambah Data Mahasiswa

**Tabel 5.3 Skenario Pengujian Fungsionalitas Menambah Data Mahasiswa**

<b>No. Pengujian</b>	<b>UJ-1.UC-02</b>
<b>Nama</b>	Menambah Data Mahasiswa
<b>Kondisi Awal</b>	Aplikasi berada pada halaman beranda Karyawan
<b>Data Uji</b>	NRP:12389397, Nama: Yudi, Alamat: Blok U, Email: <a href="mailto:yudi@gmail.com">yudi@gmail.com</a> , No Telp: 08784624, Tanggal Lahir: 4/08/1977
<b>Langkah</b>	<ol style="list-style-type: none"> <li>1. Memilih menu kelola mahasiswa</li> <li>2. Memilih Tombol tambah mahasiswa</li> <li>3. Mengisi semua isian</li> <li>4. Memilih tombol tambah</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem dapat menyimpan hasil input data mahasiswa baru
<b>Hasil yang diperoleh</b>	Sistem berhasil menyimpan hasil input mahasiswa baru
<b>Hasil Pengujian</b>	Berhasil

Tambah Mahasiswa

NRP

12389397

Nama

Yudi

Alamat

Blok U

Email

yudi@gmail.com

No Telpn

08784624

Tanggal Lahir

04/08/1977

Gambar 5.1 Halaman Tambah Mahasiswa

Mahasiswa

+

Tambah Mahasiswa

Masukan Nama Mahasiswa

NRP	Nama	Alamat	Email	No Telpn	Tanggal Lahir	#
12389397	Yudi	Blok U	yudi@gmail.com	08784624	1977-08-04	<div><div></div><div></div><div></div></div>
5109100143	Rizki Novianto				1994-05-09	<div><div></div><div></div><div></div></div>
5109100154	Henry Pandawa Putra				1994-05-19	<div><div></div><div></div><div></div></div>

Gambar 5.2 Mahasiswa Berhasil Ditambahkan

### 5.2.1.2 Pengujian Fungsionalitas Mengubah Data Mahasiswa

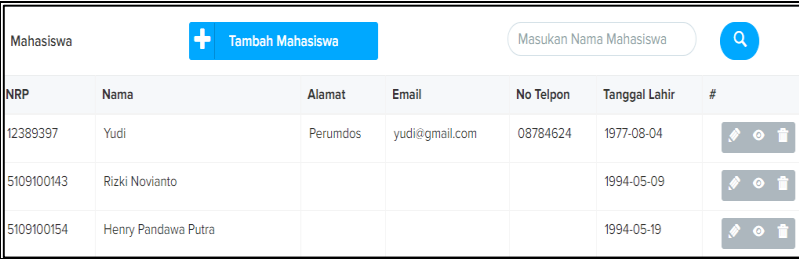
**Tabel 5.4 Skenario Pengujian Fungsionalitas Mengubah Data Mahasiswa**

<b>No. Pengujian</b>	<b>UJ-2.UC-03</b>
<b>Nama</b>	Mengubah Data Mahasiswa
<b>Kondisi Awal</b>	Aplikasi berada pada halaman kelola mahasiswa
<b>Data Uji</b>	Alamat: Perumdos
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Memilih tombol ubah pada baris mahasiswa yang ditambahkan tadi</li> <li>2. Mengubah alamat menjadi perumdos</li> <li>3. Memilih tombol ubah</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem dapat menyimpan hasil perubahan
<b>Hasil yang diperoleh</b>	Sistem berhasil menyimpan hasil perubahan
<b>Hasil Pengujian</b>	Berhasil

**Ubah Data Mahasiswa**

<b>NRP</b>	<b>No Telpn</b>
<input type="text" value="12389397"/>	<input type="text" value="08784624"/>
<b>Nama</b>	<b>Tanggal Lahir</b>
<input type="text" value="Yudi"/>	<input type="text" value="04/08/1977"/>
<b>Alamat</b>	
<input type="text" value="Perumdos"/>	
<b>Email</b>	
<input type="text" value="yudi@gmail.com"/>	

**Gambar 5.3 Halaman Mengubah Data Mahasiswa**

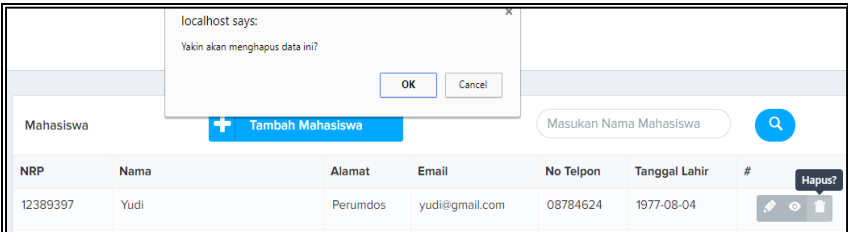


Gambar 5.4 Perubahan Berhasil Dilakukan

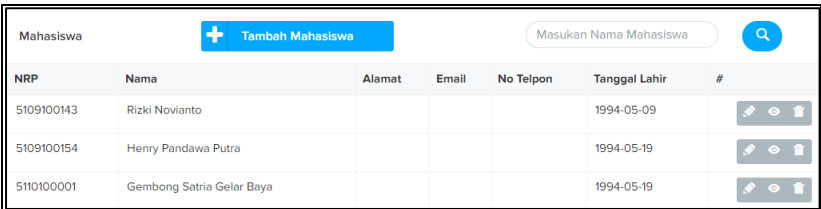
5.2.1.3 Pengujian Fungsionalitas Menghapus Data Mahasiswa

Tabel 5.5 Halaman Pengujian Fungsionalitas Menghapus Data Mahasiswa

No. Pengujian	UJ-3.UC-04
Nama	Menghapus Data Mahasiswa
Kondisi Awal	Aplikasi berada pada halaman kelola mahasiswa
Data Uji	-
Prosedur	1. Memilih tombol hapus pada baris mahasiswa yang ditambahkan pada UC-01
Hasil yang diharapkan	Sistem dapat menghapus data yang dimaksud
Hasil yang diperoleh	Sistem dapat menghapus data yang dimaksud
Hasil Pengujian	Berhasil



Gambar 5.5 Menghapus Data Mahasiswa



Gambar 5.6 Berhasil Menghapus Data Mahasiswa

5.2.1.4 Pengujian Fungsionalitas Menambah Data Dosen

Tabel 5.6 Skenario Pengujian Fungsionalitas Menambah Data Dosen

No. Pengujian	UJ-4.UC-06
Nama	Menambah Data Dosen
Kondisi Awal	Aplikasi berada pada halaman beranda Karyawan
Data Uji	NIP: 1234345, Nama: Mulyono, Kode: YM, Alamat: Perumdos Blok U, Email: yudi@gmail.com
Prosedur	1. Memilih menu kelola dosen 2. Memilih Tombol tambah dosen 3. Mengisi semua isian 4. Memilih tombol tambah
Hasil yang diharapkan	Sistem dapat menyimpan hasil input data dosen baru
Hasil yang diperoleh	Sistem berhasil menyimpan hasil input dosen baru

Hasil Pengujian	Berhasil
-----------------	----------

Tambah Dosen

NIP

1234345

Nama

Mulyono

Kode

YM

Alamat

Perumdos Blok U

Email

yudi@gmail.com

No Telpn

098080893

Tanggal Lahir

03/07/1991

Tambahkan Foto Dosen

Caril Foto

Gambar 5.7 Halaman Menambah Data Dosen

oson

Tambah Dosen

Masukan Nama Dosen

P	Kode	Nama	Alamat	Email	No Telpn	Tanggal Lahir	#
34345	YM	Mulyono	Perumdos Blok U	yudi@gmail.com	098080893	1991-07-03	<div></div> <div></div> <div></div>
2163671	DS	Dwi Sunaryono	Perumdos	dwii@if.its.ac.id	787635345	1998-10-22	<div></div> <div></div> <div></div>

Gambar 5.8 Dosen Berhasil Ditambahkan



### 5.2.1.5 Pengujian Fungsionalitas Mengubah Data Dosen

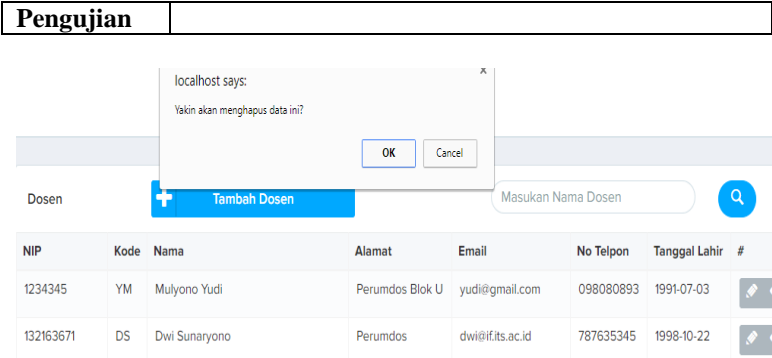
**Tabel 5.7 Skenario Pengujian Fungsionalitas Mengubah Data Dosen**

<b>No. Pengujian</b>	<b>UJ-5.UC-07</b>
<b>Nama</b>	Mengubah Data Dosen
<b>Kondisi Awal</b>	Aplikasi berada pada halaman kelola dosen
<b>Data Uji</b>	Nama: Mulyono Yudi
<b>Prosedur</b>	1. Memilih tombol ubah pada baris dosen yang ditambahkan tadi 2. Mengubah nama dosen 3. Memilih tombol ubah
<b>Hasil yang diharapkan</b>	Sistem dapat menyimpan hasil perubahan
<b>Hasil yang diperoleh</b>	Sistem berhasil menyimpan hasil perubahan
<b>Hasil Pengujian</b>	Berhasil

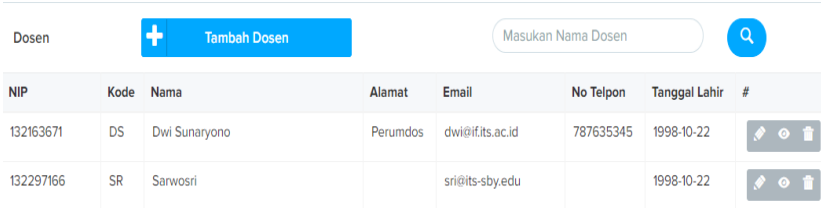
### 5.2.1.6 Pengujian Fungsionalitas Menghapus Data Dosen

**Tabel 5.8 Halaman Pengujian Fungsionalitas Menghapus Data Dosen**

<b>No. Pengujian</b>	<b>UJ-6.UC-08</b>
<b>Nama</b>	Menghapus Data Dosen
<b>Kondisi Awal</b>	Aplikasi berada pada halaman kelola dosen
<b>Prosedur</b>	1. Memilih tombol hapus pada baris dosen yang ditambahkan pada UC-04
<b>Hasil yang diharapkan</b>	Sistem dapat menghapus data yang dimaksud
<b>Hasil yang diperoleh</b>	Sistem dapat menghapus data yang dimaksud
<b>Hasil</b>	Berhasil



Gambar 5.9 Menghapus Data Dosen



Gambar 5.10 Data Dosen Berhasil Dihapus

5.2.1.7 Pengujian Fungsionalitas Menambah Data Orang Tua


Tabel 5.9 Skenario Pengujian Fungsionalitas Menambah Data Orang Tua


No. Pengujian	UJ-7.UC-26
Nama	Menambah Data Orang Tua
Kondisi Awal	Aplikasi berada pada halaman beranda Karyawan
Data Uji	Nama: Sugeng, Orang Tua dar: Achmad Fauzi Triyatno, Alamat: Perumdos, Email: <a href="mailto:sugeng@gmail.com">sugeng@gmail.com</a> , No.Telpn: 048084897
Prosedur	1. Memilih menu kelola orang tua 2. Memilih Tombol tambah orang tua 3. Mengisi semua isian


	4. Memilih tombol tambah
<b>Hasil yang diharapkan</b>	Sistem dapat menyimpan hasil input data orang tua baru
<b>Hasil yang diperoleh</b>	Sistem berhasil menyimpan hasil input orang tua baru
<b>Hasil Pengujian</b>	Berhasil


**Tambah Orang Tua**


---



 **Nama**

 **Email**

 **Orang Tua dari**

 **No Telpn**

 **Alamat**

**Gambar 5.11 Halaman Menambah Data Orang Tua**

Orang Tua

Tambah Orang Tua

Masukan Nama Orang Tua

Nama	Alamat	Email	No Telpn	Orang tua dari	#
Bapak		bapak@gmail.com		Mohammad Shahbana Satriawan	<div><div></div><div></div><div></div></div>
Sugeng	Perumdos	sugeng@gmail.com	048084897	Achmad Fauzi Triyanto	<div><div></div><div></div><div></div></div>

**Gambar 5.12 Data Orang Tua Berhasil Ditambahkan**

### 5.2.1.8 Pengujian Fungsionalitas Mengubah Data Orang Tua

**Tabel 5.10 Skenario Pengujian Fungsionalitas Mengubah Data Orang Tua**

<b>No. Pengujian</b>	<b>UJ-8.UC-27</b>
<b>Nama</b>	Mengubah Data Orang Tua
<b>Kondisi Awal</b>	Aplikasi berada pada halaman kelola orang tua
<b>Data Uji</b>	Alamat: Keputih
<b>Prosedur</b>	1. Memilih tombol ubah pada baris orang tua yang ditambahkan tadi 2. Mengubah Alamat 3. Memilih tombol ubah
<b>Hasil yang diharapkan</b>	Sistem dapat menyimpan hasil perubahan
<b>Hasil yang diperoleh</b>	Sistem berhasil menyimpan hasil perubahan
<b>Hasil Pengujian</b>	Berhasil

Ubah Data Orang Tua

Nama

Sugeng

Orang Tua dari

Achmad Fauzi Triyanto

Alamat

Keputih

Email

sugeng@gmail.com

No Telpn

048084897

Gambar 5.13 Halaman Ubah Data Orang Tua

Orang Tua		<div><div></div>Tambah Orang Tua</div>		<div>Masukan Nama Orang Tua</div>		<div><div></div></div>
Nama	Alamat	Email	No Telpn	Orang tua dari	#	
Lely Fadriaty	Jln Indrakila 10 A Surabaya	lely@gmail.com		Mohammad Shahbana Satriawan	<div><div></div><div></div><div></div></div>	
Sugeng	Keputih	sugeng@gmail.com	048084897	Achmad Fauzi Triyanto	<div><div></div><div></div><div></div></div>	

Gambar 5.14 Berhasil Ubah Data Orang Tua

Ubah Data Dosen

NIP

1234345

Nama

Mulyono Yudi

Kode

YM

Alamat

Perumdos Blok U

Email

yudi@gmail.com

No Telpn

098080893

Tanggal Lahir

03/07/1991

Tambahkan Foto Dosen

Carli Foto

Gambar 5.15 Halaman Ubah Dosen

Dosen

Tambah Dosen

Masukan Nama Dosen

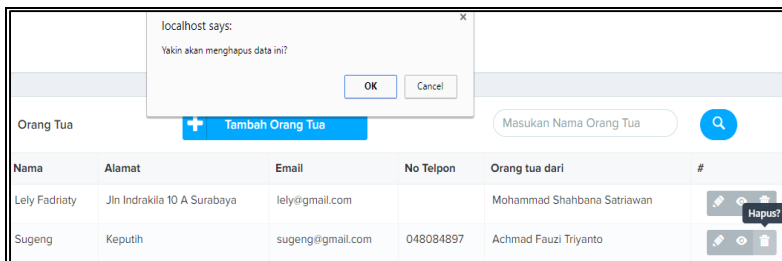
NIP	Kode	Nama	Alamat	Email	No Telpn	Tanggal Lahir	#
1234345	YM	Mulyono Yudi	Perumdos Blok U	yudi@gmail.com	098080893	1991-07-03	<div></div>
132163671	DS	Dwi Sunaryono	Perumdos	dwi@iif.its.ac.id	787635345	1998-10-22	<div></div>

Gambar 5.16 Berhasil Menyimpan Perubahan Data Dosen

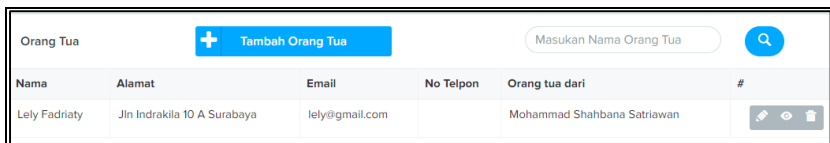
### 5.2.1.9 Pengujian Fungsionalitas Menghapus Data Orang Tua

**Tabel 5.11 Halaman Pengujian Fungsionalitas Menghapus Data Orang Tua**

<b>No. Pengujian</b>	<b>UJ-9.UC-28</b>
<b>Nama</b>	Menghapus Data Orang Tua
<b>Kondisi Awal</b>	Aplikasi berada pada halaman kelola orang tua
<b>Prosedur</b>	1. Memilih tombol hapus pada baris orang tua yang ditambahkan tadi
<b>Hasil yang diharapkan</b>	Sistem dapat menghapus data yang dimaksud
<b>Hasil yang diperoleh</b>	Sistem dapat menghapus data yang dimaksud
<b>Hasil Pengujian</b>	Berhasil



**Gambar 5.17 Menghapus Data Orang Tua**



**Gambar 5.18 Data Orang Tua Berhasil Dihapus**

5.2.1.10 Pengujian Fungsionalitas Menambah Data Ruang

Tabel 5.12 Skenario Pengujian Fungsionalitas Menambah Data Ruang

No. Pengujian	UJ-10.UC-14
Nama	Menambah Data Ruang
Kondisi Awal	Aplikasi berada pada halaman beranda Karyawan
Data Uji	Kode: IF-302, Nama: Lab Rekayasa Perangkat Lunak
Prosedur	1. Memilih menu kelola ruangan 2. Memilih Tombol tambah ruangan 3. Mengisi semua isian 4. Memilih tombol tambah
Hasil yang diharapkan	Sistem dapat menyimpan hasil input data ruangan baru
Hasil yang diperoleh	Sistem berhasil menyimpan hasil input ruangan baru
Hasil Pengujian	Berhasil

Tambah Ruang

Kode

IF 302

Nama

Lab Rekayasa Perangkat Lunak

Longitude

0

Latitude

0

Altitude













0

+

↺

Gambar 5.19 Halaman Menambah Data Ruang



ALPRO	Algoritma Pemrograman	112.797242	-7.27979	15	  
MI	Manajemen Informasi	112.797149	-7.279789	15	  
DTK	Dasar dan Terapan Komunikasi	112.79711	-7.279785	15	  
IF 302	Lab Rekayasa Perangkat Lunak	0	0	0	  

**Gambar 5.20 Berhasil Menambah Data Ruangan**

### 5.2.1.11 Pengujian Fungsionalitas Mengubah Data Ruangan

**Tabel 5.13 Skenario Pengujian Fungsionalitas Mengubah Data Ruangan**

<b>No. Pengujian</b>	<b>UJ-11.UC-15</b>
<b>Nama</b>	Mengubah Data Ruangan
<b>Kondisi Awal</b>	Aplikasi berada pada halaman kelola ruangan
<b>Data Uji</b>	Nama: Lab RPL
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Memilih tombol ubah pada baris ruangan yang ditambahkan tadi</li> <li>2. Mengubah nama</li> <li>3. Memilih tombol ubah</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem dapat menyimpan hasil perubahan
<b>Hasil yang diperoleh</b>	Sistem berhasil menyimpan hasil perubahan
<b>Hasil Pengujian</b>	Berhasil

Ubah Data Ruangan

Kode

IF 302

Nama

Lab RPL

Longitude

0

Latitude

0

Altitude

0

+

↺

Gambar 5.21 Halaman Mengubah Data Ruangan

MI	Manajemen Informasi	112.797149	-7.279789	15	<div><div></div><div></div><div></div></div>
DTK	Dasar dan Terapan Komunikasi	112.79711	-7.279785	15	<div><div></div><div></div><div></div></div>
IF 302	Lab RPL	0	0	0	<div><div></div><div></div><div></div></div>

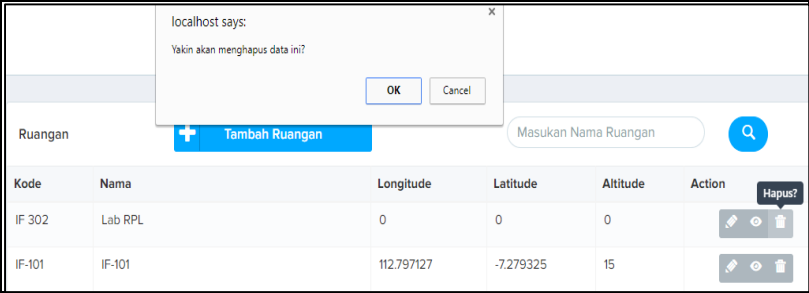
Gambar 5.22 Berhasil Mengubah Data Ruangan

5.2.1.12 Pengujian Fungsionalitas Menghapus Data Ruangan

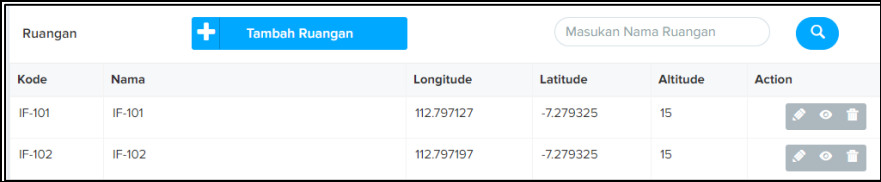
Tabel 5.14 Fungsionalitas Menghapus Data Ruangan

No. Pengujian	UJ-12.UC-15
Nama	Menghapus Data Ruangan
Kondisi Awal	Aplikasi berada pada halaman kelola ruangan
Prosedur	1. Memilih tombol hapus pada baris ruangan yang ditambahkan tadi
Hasil yang	Sistem dapat menghapus data yang dimaksud

diharapkan	
Hasil yang diperoleh	Sistem dapat menghapus data yang dimaksud
Hasil Pengujian	Berhasil



Gambar 5.23 Halaman Menghapus Data Ruangan



Gambar 5.24 Berhasil Menghapus Data Ruangan

5.2.1.13 Pengujian Fungsionalitas Menambah Data Matakuliah

Tabel 5.15 Skenario Pengujian Fungsionalitas Menambah Data Matakuliah

No. Pengujian	UJ-13.UC-10
Nama	Menambah Data Matakuliah
Kondisi Awal	Aplikasi berada pada halaman beranda Karyawan
Data Uji	Nama: Dasar Pemrograman, Kode: Dasprog, Kode Integra: KI141301, Jumlah SKS: 4, RMK: AP, Semester: 1

<b>Prosedur</b>	1. Memilih menu kelola matakuliah 2. Memilih Tombol tambah matakuliah 3. Mengisi semua isian 4. Memilih tombol tambah
<b>Hasil yang diharapkan</b>	Sistem dapat menyimpan hasil input data matakuliah baru
<b>Hasil yang diperoleh</b>	Sistem berhasil menyimpan hasil input matakuliah baru
<b>Hasil Pengujian</b>	Berhasil

**Tambah Matkul**

**Nama**  
Dasar Pemrograman

**Kode**  
Dasprog

**Jumlah SKS**  
4

**RMK**  
AP

**Semester**  
1

+ ↺

Gambar 5.25 Halaman Menambah Matakuliah

Mata Kuliah + Tambah Mata Kuliah Masukan Nama Mata Kuliah

Kode Integra	Nama Matakuliah	SKS	RMK	Semester	
KI141301	Dasar Pemrograman	4		1	
KI141303	Sistem dan Teknologi Informasi	2		1	

Gambar 5.26 Berhasil Menambahkan Matakuliah

### 5.2.1.14 Pengujian Fungsionalitas Mengubah Data Matakuliah

**Tabel 5.16 Skenario Pengujian Fungsionalitas Mengubah Data Matakuliah**

<b>No. Pengujian</b>	<b>UJ-13.UC-15</b>
<b>Nama</b>	Mengubah Data Matakuliah
<b>Kondisi Awal</b>	Aplikasi berada pada halaman kelola matakuliah
<b>Data Uji</b>	Nama: Dasar Pemrograman Terstruktur
<b>Prosedur</b>	1. Memilih tombol ubah pada baris matakuliah yang ditambahkan tadi 2. Mengubah nama matakuliah 3. Memilih tombol ubah
<b>Hasil yang diharapkan</b>	Sistem dapat menyimpan hasil perubahan
<b>Hasil yang diperoleh</b>	Sistem berhasil menyimpan hasil perubahan
<b>Hasil Pengujian</b>	Berhasil

Update Matkul

Nama

Dasar Pemrograman Terstruktur

Jumlah SKS

4

Kode

DASPROG

RMK

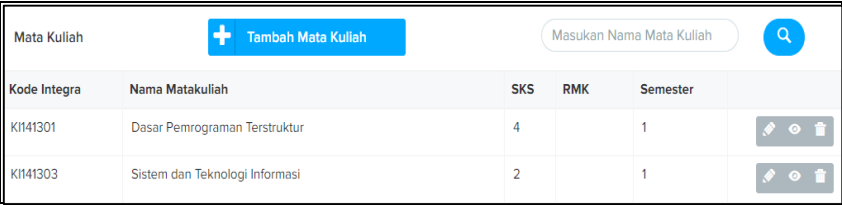
Semester

1

+

↺

**Gambar 5.27 Halaman Mengubah Matakuliah**

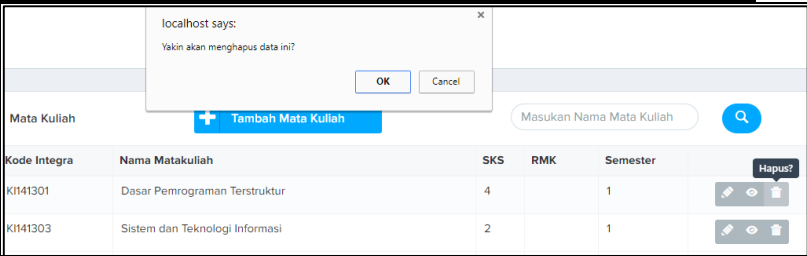


Gambar 5.28 Berhasil Mengubah Data Matakuliah

5.2.1.15 Pengujian Fungsionalitas Menghapus Data Matakuliah

Tabel 5.17 Halaman Pengujian Fungsionalitas Menghapus Data

No. Pengujian	UJ-15.UC-16
Nama	Menghapus Data Matakuliah
Kondisi Awal	Aplikasi berada pada halaman kelola matakuliah
Prosedur	1. Memilih tombol hapus pada baris matakuliah yang ditambahkan tadi
Hasil yang diharapkan	Sistem dapat menghapus data yang dimaksud
Hasil yang diperoleh	Sistem dapat menghapus data yang dimaksud
Hasil Pengujian	Berhasil



Gambar 5.29 Halaman Menghapus Data Matakuliah

### 5.2.1.16 Pengujian Fungsionalitas Menambah Data Kelas

**Tabel 5.18 Skenario Pengujian Fungsionalitas Menambah Data Orang Tua**

<b>No. Pengujian</b>	<b>UJ-16.UC-18</b>
<b>Nama</b>	Menambah Data Kelas
<b>Kondisi Awal</b>	Aplikasi berada pada halaman beranda Karyawan
<b>Data Uji</b>	Mata Kuliah: Pemrograman Perangkat Bergerak, , Kode Kelas: A, Dosen Pengajar: Dwi Sunaryono, Hari: Kamis, Waktu: 13.30, Ruangan: Lab Pemrograman 2
<b>Prosedur</b>	1. Memilih menu kelola kelas 2. Memilih Tombol tambah kelas 3. Mengisi semua isian 4. Memilih tombol tambah
<b>Hasil yang diharapkan</b>	Sistem dapat menyimpan hasil input data kelas baru
<b>Hasil yang diperoleh</b>	Sistem berhasil menyimpan hasil input kelas baru
<b>Hasil Pengujian</b>	Berhasil

Tambah Kelas

Mata Kuliah

Pemrograman Perangkat Bergerak

Kode Kelas

A

Dosen Pengajar

Dwi Sunaryono

Dosen Pengajar Kedua (Opsional)

Hari

Kamis

Waktu

13:30:00

Ruangan

Lab Pemrograman 2

Hari Kedua(Opsional)

Waktu (Opsional)

Activate Windows

Gambar 5.30 Halaman Menambah Kelas

PBKK	A	ALPRO	Senin,09:30:00-11:30:00	Dwi Sunaryono				
PPB	A	LP2	Kamis,13:30:00-16:30:00	Dwi Sunaryono				

Gambar 5.31 Berhasil Menambah Kelas



### 5.2.1.17 Pengujian Fungsionalitas Mengubah Data Kelas

**Tabel 5.19 Skenario Pengujian Fungsionalitas Mengubah Data Orang Tua**

<b>No. Pengujian</b>	<b>UJ-17.UC-19</b>
<b>Nama</b>	Mengubah Data Kelas
<b>Kondisi Awal</b>	Aplikasi berada pada halaman kelola kelas
<b>Data Uji</b>	Dosen Pengajar: Abdul Munif
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Memilih tombol ubah pada baris kelas yang ditambahkan tadi</li> <li>2. Mengubah data Dosen Pengajar</li> <li>3. Memilih tombol ubah</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem dapat menyimpan hasil perubahan
<b>Hasil yang diperoleh</b>	Sistem berhasil menyimpan hasil perubahan
<b>Hasil Pengujian</b>	Berhasil

**Gambar 5.32 Halaman Mengubah Data Kelas**

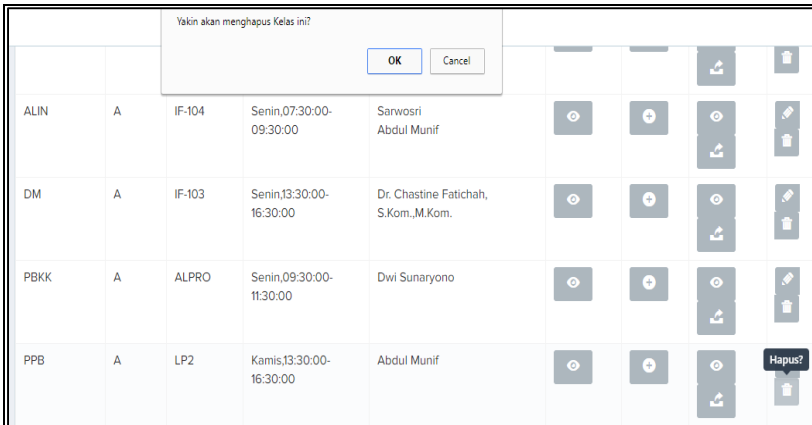
PBKK	A	ALPRO	Senin,09:30:00-11:30:00	Dwi Sunaryono			 	
PPB	A	LP2	Kamis,13:30:00-16:30:00	Abdul Munif			 	

Gambar 5.33 Berhasil Mengubah Data Kelas

5.2.1.18 Pengujian Fungsionalitas Menghapus Data Matakuliah

Tabel 5.20 Halaman Pengujian Fungsionalitas Menghapus Data

No. Pengujian	UJ-18.UC-19
Nama	Menghapus Data Kelas
Kondisi Awal	Aplikasi berada pada halaman kelola kelas
Prosedur	1. Memilih tombol hapus pada baris kelas yang ditambahkan tadi
Hasil yang diharapkan	Sistem dapat menghapus data yang dimaksud
Hasil yang diperoleh	Sistem dapat menghapus data yang dimaksud
Hasil Pengujian	Berhasil



**Gambar 5.34 Halaman Menghapus Data Kelas**

### 5.2.1.19 Pengujian Fungsionalitas Menambah Peserta Kelas

**Tabel 5.21 Halaman Pengujian Fungsionalitas Menghapus Data**

<b>No. Pengujian</b>	<b>UJ-19.UC-29</b>
<b>Nama</b>	Menambah Peserta Kelas
<b>Kondisi Awal</b>	Aplikasi berada pada halaman kelola kelas
<b>Data Uji</b>	File Excel berisi nrp 5113100146, 5113100050, 5113100001, 511310003, 5113100112
<b>Prosedur</b>	1. Memilih tombol ubah pada baris kelas yang ditambahkan tadi 2. Mengupload file excel berisi peserta kelas 3. Memilih tombol upload
<b>Hasil yang diharapkan</b>	Sistem dapat menambahkan peserta kelas
<b>Hasil yang diperoleh</b>	Sistem berhasil menambahkan peserta kelas
<b>Hasil Pengujian</b>	Berhasil

no	nrp
1	5113100146
2	5113100050
3	5113100001
4	511310003
5	5113100112

Gambar 5.35 Isi File Excel Contohupload.xls

Format Upload

Lihat Format

File

Choose File

contohupload (3).xlsx

Upload File

Gambar 5.36 Halaman Upload File

Kelas PPB A																		
			Pertemuan															
No	NRP	Nama	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	5113100001	Muhammad Adnan Yusuf																
2	5113100050	Mohammad Shahbana Satriawan																
3	5113100112	Ronald Gunawan Radityatama																
4	5113100146	Kevin Zulkarnain Yuseti																

Gambar 5.37 Peserta Kelas Berhasil Ditambahkan

### 5.2.1.20 Pengujian Fungsionalitas Melihat Detail Perkuliahan

Tabel 5.22 Halaman Pengujian Fungsionalitas Menghapus Data

<b>No. Pengujian</b>	<b>UJ-20.UC-21</b>
<b>Nama</b>	Melihat Detail Perkuliahan
<b>Kondisi Awal</b>	Aplikasi berada pada halaman kelola kelas
<b>Data Uji</b>	Kelas Basis Data Terdistribusi A
<b>Prosedur</b>	1. Memilih tombol perkuliahan pada baris suatu kelas
<b>Hasil yang diharapkan</b>	Sistem dapat menampilkan detail perkuliahan
<b>Hasil yang diperoleh</b>	Sistem berhasil menampilkan detail perkuliahan
<b>Hasil Pengujian</b>	Berhasil

Perkuliahan Kelas Basis Data Terdistribusi A						
Pertemuan	Ruangan	Tanggal	Jadwal	Status Kelas	Status Jadwal	Rekap Kehadiran
1	IF-106	2017-02-09	Kamis,13:30:00-16:30:00	Selesai	Normal	
2	IF-106	2017-02-16	Kamis,13:30:00-16:30:00	Selesai	Pengganti	
3	IF-106	2017-02-23	Kamis,13:30:00-16:30:00	Aktif	Normal	
4	IF-106	2017-03-02	Kamis,13:30:00-16:30:00	Belum Aktif	Normal	

Gambar 5.38 Melihat Detail Perkuliahan

5.2.1.21 Pengujian Fungsionalitas Melihat Rekap Kehadiran

Tabel 5.23 Halaman Pengujian Fungsionalitas Melihat Rekap Kehadiran

No. Pengujian	UJ-21.UC-23
Nama	Melihat Rekap Kehadiran
Kondisi Awal	Penguji berada pada halaman kelola kelas
Prosedur	1. Memilih tombol Rekap Kehadiran pada baris suatu kelas
Hasil yang diharapkan	Sistem dapat menampilkan rekap kehadiran
Hasil yang diperoleh	Sistem berhasil menampilkan rekap kehadiran
Hasil Pengujian	Berhasil

Kelas ERPL A																		
No	NRP	Nama	Pertemuan															
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	5111100138	Sandri Ayunier K	M	M	I	A	M	I	M	M	M	M	M	M	M	M	M	A
2	5111100198	Hery Nugroho	M	M	M	M	M	I	M	A	M	M	M	M	M	M	M	M
3	5112100186	Andre Abdirrosyid	M	M	M	M	A	I	M	M	M	M	I	M	M	M	M	M

Gambar 5.39 Halaman Melihat Rekap Kehadiran

## 5.2.2 Pengujian Fungsionalitas Pada Aktor Dosen

Pengujian Fungsionalitas pada aktor dosen meliputi dua aplikasi, yakni Aplikasi Web dan Perangkat bergerak

### 5.2.2.1 Pengujian Fungsionalitas Mengaktifkan Perkuliahan

**Tabel 5.24 Skenario Pengujian Fungsionalitas Mengaktifkan Perkuliahan**

<b>No. Pengujian</b>	<b>UJ-22.UC-36</b>
<b>Nama</b>	Mengaktifkan Perkuliahan
<b>Kondisi Awal</b>	Dosen belum mengaktifkan kelas
<b>Data Uji</b>	Perkuliahan BDT A (Mobile) Perkuliahan ERPL A (Web)
<b>Prosedur</b>	1. Memilih tombol aktifkan kelas
<b>Hasil yang diharapkan</b>	Status kelas berubah menjadi aktif
<b>Hasil yang diperoleh</b>	Sistem kelas berhasil berubah menjadi aktif
<b>Hasil Pengujian</b>	Berhasil

Jadwal Perkuliahan Anda Hari Ini						
Kode Mata Kuliah	Kode Kelas	Pertemuan Ruangan	Ke	Jadwal	Dosen Pengajar	Status
BDT	A	IF-102	16	Jumat, 23:02:00	Dwi Sunaryono Sarwosri	Belum Aktif

**Gambar 5.40 Halaman Pengaktifan Matakuliah (Web)**

Jadwal Perkuliahan Anda Hari Ini						
Kode Mata Kuliah	Kode Kelas	Ruangan	Pertemuan Ke	Jadwal	Dosen Pengajar	Status
BDT	A	IF-102	16	Jumat, 23:02:00	Dwi Sunaryono Sarwosri	Aktif

**Gambar 5.41 Perkuliahan Berhasil Diaktifkan (Web)**



**Gambar 5.42 Halaman Mengaktifkan Kuliah (Mobile)**



**Gambar 5.43 Perkuliahan Berhasil Diaktifkan (Mobile)**



5.2.2.2 Pengujian Fungsionalitas Mengakhiri Kelas

Tabel 5.25 Skenario Pengujian Fungsionalitas Mengakhiri Kelas

No. Pengujian	UJ-23.UC-36
Nama	Mengakhiri Kelas
Kondisi Awal	Dosen belum mengakhiri kelas
Data Uji	Perkuliahahan BDT A (Mobile) Perkuliahan ERPL A (Web)
Prosedur	Memilih tombol akhiri kelas
Hasil yang diharapkan	Status kelas berubah menjadi berakhir
Hasil yang diperoleh	Status kelas berhasil berubah menjadi berakhir
Hasil Pengujian	Berhasil



Gambar 5.44 Halaman Mengakhiri Kelas



Gambar 5.45 Berhasil Mengakhiri Kelas (Web)



**Gambar 5.46 Berhasil Mengakhiri Kelas (Mobile)**

### **5.2.2.3 Pengujian Fungsionalitas Melihat Detail Perkuliahan**

**Tabel 5.26 Skenario Pengujian Fungsionalitas Melihat Detail Perkuliahan**

<b>No. Pengujian</b>	<b>UJ-24.UC-21</b>
<b>Nama</b>	Melihat Detail Perkuliahan
<b>Kondisi Awal</b>	Dosen belum memilih perkuliahan
<b>Data Uji</b>	Perkuliahan Kelas BDT A
<b>Prosedur</b>	Memilih tombol detail perkuliahan
<b>Hasil yang diharapkan</b>	Sistem menampilkan Detail Perkuliahan
<b>Hasil yang diperoleh</b>	Sistem berhasil menampilkan Detail Perkuliahan
<b>Hasil Pengujian</b>	Berhasil

Daftar Kelas yang anda ampu

Kode Mata Kuliah	Kode Kelas	Ruangan	Jadwal	Dosen Pengajar	Detail	Rekap
					Perkuliahan	Absensi
BDT	A	IF-106	Kamis,13:30:00-16:30:00	Dwi Sunaryono Sanwosri		
ERPL	A	IF-101	Selasa,13:30:00-16:30:00	Dwi Sunaryono Sanwosri		

**Gambar 5.47 Halaman Melihat Detail Perkuliahan (Web)**

Perkuliahan Kelas Basis Data Terdistribusi A

Pertemuan	Ruangan	Tanggal	Jadwal	Status Kelas	Status Jadwal	Rekap Kehadiran
1	IF-106	2017-02-09	Kamis,13:30:00-16:30:00	Selesai	Normal	
2	IF-106	2017-02-16	Kamis,13:30:00-16:30:00	Selesai	Pengganti	
3	IF-106	2017-02-23	Kamis,13:30:00-16:30:00	Aktif	Normal	
4	IF-106	2017-03-02	Kamis,13:30:00-16:30:00	Belum Aktif	Normal	

**Gambar 5.48 Berhasil Menampilkan Detail Perkuliahan (Web)**



**Gambar 5.49 Berhasil Menampilkan Detail Perkuliahan (Mobile)**

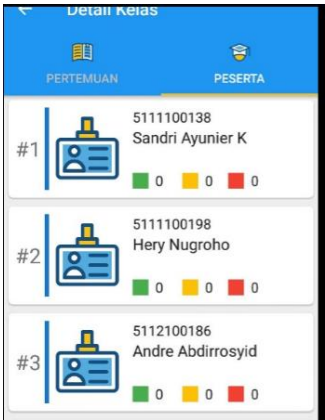
#### **5.2.2.4 Pengujian Fungsionalitas Melihat Rekap Kehadiran Kelas**

**Tabel 5.27 Skenario Pengujian Fungsionalitas Melihat Rekap Kehadiran Kelas**

<b>No. Pengujian</b>	<b>UJ-25.UC-23</b>
<b>Nama</b>	Melihat Rekap Kehadiran kelas
<b>Kondisi Awal</b>	Dosen berada pada halaman list kelas diampu
<b>Data Uji</b>	Kelas ERPL A
<b>Prosedur</b>	Memilih tombol rekap kehadiran
<b>Hasil yang diharapkan</b>	Sistem dapat menampilkan rekap kehadiran mahasiswa
<b>Hasil yang diperoleh</b>	Sistem berhasil menampilkan rekap kehadiran mahasiswa
<b>Hasil Pengujian</b>	Berhasil

Kelas ERPL A																		
			Pertemuan															
No	NRP	Nama	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	5111100138	Sandri Ayunier K	M	M	I	A	M	I	M	M	M	M	M	M	M	M	M	A
2	5111100198	Hery Nugroho	M	M	M	M	M	I	M	A	M	M	M	M	M	M	M	M
3	5112100186	Andre Abdirrosyid	M	M	M	M	A	I	M	M	M	M	I	M	M	M	M	M

Gambar 5.50 Berhasil Menampilkan Rekap Kehadiran (Web)



Gambar 5.51 Berhasil Menampilkan Rekap Kehadiran (Mobile)

5.2.2.5 Pengujian Fungsionalitas Mengubah Status Kehadiran Mahasiswa

Tabel 5.28 Skenario Pengujian Fungsionalitas Mengubah Status Kehadiran Mahasiswa

No. Pengujian	UJ-26.UC-23
Nama	Mengganti Status Kehadiran Mahasiswa
Kondisi Awal	Dosen berada pada halaman detail pertemuan suatu perkuliahan

<b>Data Uji</b>	Status Kehadiran Mohammad Shahbana Satriawan pada perkuliahan BDT A pertemuan 16 (Web), Status Kehadiran Andre Abdirosyid pada perkuliahan ERPL A pertemuan 1
<b>Prosedur</b>	Memilih icon ubah, pada baris mahasiswa. Pilih Hadir, Ijin atau Absen. Kemudian pilih tombol "PILIH"
<b>Hasil yang diharapkan</b>	Sistem mengubah kolom ket_kehadiran pada perkuliahan dan mahasiswa yang dipilih
<b>Hasil yang diperoleh</b>	Sistem berhasil mengubah kolom ket_kehadiran pada perkuliahan dan mahasiswa yang dipilih
<b>Hasil Pengujian</b>	Berhasil

Ubah Kehadiran Mahasiswa

**Kelas**

**Nama Mahasiswa**

**Kode**

**Status Kehadiran**

M
▼

**Pertemuan**

**Gambar 5.52 Halaman Ubah Status Kehadiran Mahasiswa (Web)**

Ubah Status Kehadiran dari peserta 5112100186 - Andre...

☒ Hadir

☐ Ijin

☐ Absen

BATAL PILIH

**Gambar 5.53 Ubah Status Kehadiran Mahasiswa (Mobile)**

### 5.2.2.6 Pengujian Fungsionalitas Mengubah Jadwal Perkuliahan

**Tabel 5.29 Skenario Pengujian Fungsionalitas Mengubah Jadwal Perkuliahan**

<b>No. Pengujian</b>	<b>UJ-27.UC-22</b>
<b>Nama</b>	Mengubah Jadwal Perkuliahan
<b>Kondisi Awal</b>	Dosen berada pada halaman detail kelas
<b>Data Uji</b>	Perkuliahan ERPL pertemuan 16, tanggal pengganti: 15/06/2017, waktu: 13:30:00, Ruangan: Algoritma Pemrograman(Web). Perkuliahan ERPL pertemuan 16, tanggal pengganti: 15/06/2017, waktu: 13:30:00, Ruangan: Algoritma Pemrograman.
<b>Prosedur</b>	Memilih tombol "Sementara" pada ubah jadwal. Mengisi jadwal baru dan ruangan
<b>Hasil yang diharapkan</b>	Sistem dapat menyimpan perubahan
<b>Hasil yang diperoleh</b>	Sistem berhasil menyimpan perubahan
<b>Hasil Pengujian</b>	Berhasil

**Ganti Jadwal**

<p><b>Nama</b></p> <p>Ekonomi Rekayasa Perangkat Lunak</p>	<p><b>Pilih tanggal pengganti</b></p> <p>15/06/2017</p>
<p><b>Kode</b></p> <p>A</p>	<p><b>Waktu</b></p> <p>13:30:00</p>
<p><b>Pertemuan</b></p> <p>16</p>	<p><b>Ruangan</b></p> <p>Algoritma Pemrograman</p>

+
G

**Gambar 5.54 Halaman Pengubahan Jadwal Perkuliahan (Web)**

**Penjadwalan Ulang**

**Ganti Jadwal**

▼ ✓ ✓ ✓ 4

Tanggal Waktu Ruangan **Konfirmasi**

---

**Perubahan Jadwal**

Pertemuan ke- 1

**Pemrograman berbasis Kerangka Kerja**

Lokasi

Ruangan

**Algoritma Pemrograman**

Kelas

**A**

Ruangan Baru

**Algoritma Pemrograman**

Waktu Perkuliahan

📅 **Senin, 06 Feb 2017**

**Penjadwalan Ulang**

**Kerangka Kerja**

Lokasi

Ruangan

**Algoritma Pemrograman**

Kelas

**A**

Ruangan Baru

**Algoritma Pemrograman**

Waktu Perkuliahan

📅 **Senin, 06 Feb 2017**

↓

📅 **Jumat, 23 Jun 2017**

mulai

🕒 **09:30 AM**

selesai

🕒 **11:30 AM**

➡

mulai

🕒 **01:30 PM**

selesai

🕒 **04:30 PM**

BATALKAN PERUBAHAN

SIMPAN PERUBAHAN

**Gambar 5.55 Halaman Pengubahan Jadwal Perkuliahan (Mobile)**



### 5.2.3 Pengujian Fungsionalitas Pada Aktor Mahasiswa

Pengujian Fungsionalitas pada aktor mahasiswa hanya meliputi aplikasi perangkat bergerak

#### 5.2.3.1 Pengujian Fungsionalitas Melihat Perkuliahan Aktif Hari Ini

**Tabel 5.30 Melihat Perkuliahan Aktif Hari Ini**

<b>No. Pengujian</b>	<b>UJ-28.UC-21</b>
<b>Nama</b>	Melihat Perkuliahan Aktif Hari Ini
<b>Kondisi Awal</b>	Mahasiswa belum berada di halaman perkuliahan aktif
<b>Data Uji</b>	Perkuliahan ERPL pertemuan 16
<b>Prosedur</b>	Memilih tab perkuliahan aktif
<b>Hasil yang diharapkan</b>	Sistem menampilkan perkuliahan aktif hari ini
<b>Hasil yang diperoleh</b>	Sistem berhasil menampilkan perkuliahan aktif hari ini
<b>Hasil Pengujian</b>	Berhasil

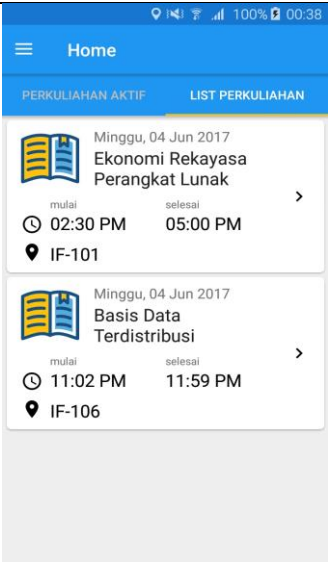


**Gambar 5.56 Halaman Perkuliahan Aktif**

5.2.3.2 Pengujian Fungsionalitas Melihat List Perkuliahan Hari Ini

Tabel 5.31 Melihat List Perkuliahan Hari Ini

No. Pengujian	UJ-29.UC-21
Nama	Melihat List Perkuliahan Hari Ini
Kondisi Awal	Mahasiswa belum berada di halaman list perkuliahan
Prosedur	Memilih tab list perkuliahan
Hasil yang diharapkan	Sistem menampilkan perkuliahan hari ini
Hasil yang diperoleh	Sistem berhasil menampilkan list perkuliahan hari ini
Hasil Pengujian	Berhasil

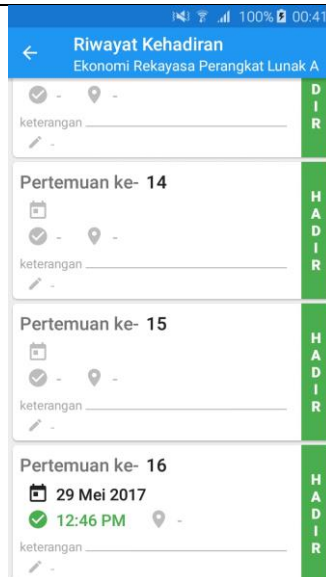


Gambar 5.57 Menampilkan List Perkuliahan

### 5.2.3.3 Pengujian Fungsionalitas Melihat Riwayat Kehadiran di suatu kelas

Tabel 5.32 Melihat Riwayat Kehadiran di suatu kelas

<b>No. Pengujian</b>	UJ-30.UC-23
<b>Nama</b>	Melihat Riwayat Kehadiran di suatu kelas
<b>Kondisi Awal</b>	Mahasiswa berada di halaman jadwal kelas
<b>Data Uji</b>	Riwayat Kehadiran Mahasiswa NRP 5113100050 pada perkuluahan ERPL A
<b>Prosedur</b>	Memilih salah satu kelas
<b>Hasil yang diharapkan</b>	Sistem menampilkan riwayat kehadiran
<b>Hasil yang diperoleh</b>	Sistem berhasil menampilkan riwayat kehadiran
<b>Hasil Pengujian</b>	Berhasil



Gambar 5.58 Menampilkan Riwayat Kehadiran

## 5.2.4 Pengujian Fungsionalitas Pada Aktor Orang Tua

Pengujian Fungsionalitas pada aktor dosen meliputi dua aplikasi, yakni Aplikasi Web dan Perangkat bergerak

### 5.2.4.1 Pengujian Fungsionalitas Melihat Perkuliahan Anak Minggu Ini

**Tabel 5.33 Melihat Perkuliahan Anak Minggu Ini**

<b>No. Pengujian</b>	<b>UJ-31.UC-21</b>
<b>Nama</b>	Melihat Perkuliahan Anak Minggu Ini
<b>Kondisi Awal</b>	Orang Tua tidak berada dalam halaman beranda
<b>Prosedur</b>	Memilih halaman beranda
<b>Data Uji</b>	Perkuliahan Mahasiswa NRP 5113100050 pada minggu ke 16
<b>Hasil yang diharapkan</b>	Sistem menampilkan perkuliahan anak minggu ini
<b>Hasil yang diperoleh</b>	Sistem berhasil menampilkan perkuliahan anak minggu ini
<b>Hasil Pengujian</b>	Berhasil

Jadwal Kuliah Mohammad Shahbana Satriawan Minggu Ini							
Mata Kuliah	Kelas	Pertemuan	Tanggal	Jadwal	Dosen	Status Perkuliahan	Ket Kehadiran
Ekonomi Rekayasa Perangkat Lunak	A	16	2017-06-05	Jumat, 14:30:00-17:00:00	Dwi Sunaryono Sarwosri	Belum Aktif	M
Basis Data Terdistribusi	A	16	2017-06-05	Jumat, 23:02:00-23:59:00	Dwi Sunaryono Sarwosri	Selesai	M

**Gambar 5.59 Melihat Perkuliahan Anak Minggu Ini**

### 5.2.4.2 Pengujian Fungsionalitas Monitor Kehadiran Anak

**Tabel 5.34 Monitor Kehadiran Anak**

<b>No. Pengujian</b>	<b>UJ-32.UC-23</b>
<b>Nama</b>	Monitor Kehadiran Anak
<b>Kondisi Awal</b>	Orang Tua tidak berada dalam halaman monitor anak
<b>Data Uji</b>	Rekap Kehadiran Mahasiswa NRP 5113100050 pada perkuliahan semester ini
<b>Prosedur</b>	Memilih halaman monitor anak
<b>Hasil yang diharapkan</b>	Sistem menampilkan rekap kehadiran anak di semua kelas semester ini
<b>Hasil yang diperoleh</b>	Sistem berhasil menampilkan rekap kehadiran anak di semua kelas semester ini
<b>Hasil Pengujian</b>	Berhasil

Monitor Kuliah Mohammad Shahbana Satriawan Semester ini

Mata Kuliah	Kelas	Jadwal	Masuk	Ijin	Absen	Perkuliahan
Ekonomi Rekayasa Perangkat Lunak	A	Selasa,13:30:00-16:30:00	13	1	1	
Basis Data Terdistribusi	A	Kamis,13:30:00-16:30:00	12	2	1	
Aljabar Linier	A	Senin,07:30:00-09:30:00	11	3	1	
Pemrograman Perangkat Bergerak	A	Kamis,13:30:00-16:30:00	10	4	1	

**Gambar 5.60 Halaman Monitor Anak**

5.2.5 Pengujian Fungsionalitas Notifikasi

5.2.5.1 Pengujian Fungsionalitas Notifikasi Ketidakhadiran mahasiswa pada Perkuliahan

No. Pengujian	UJ-33.UC-35
Nama	Notifikasi Ketidakhadiran Mahasiswa pada Perkuliahan
Kondisi Awal	Dosen mengganti status kehadiran seorang mahasiswa menjadi Izin
Data Uji	Mahasiswa: Mohammad Shahbana Satriawan. Perkuliahan ERPL A, Pertemuan: 16, Ket Kehadiran”: I, Pesan: Sakit
Prosedur	Dosen mengganti status kehadiran mahasiswa sesuai dengan data uji
Hasil yang diharapkan	Sistem mengirimkan notifikasi ke perangkat milik orangtua mahasiswa
Hasil yang diperoleh	Sistem berhasil mengirimkan notifikasi ke perangkat milik orangtua mahasiswa
Hasil Pengujian	Berhasil

Ubah Kehadiran Mahasiswa

Kelas

Ekonomi Rekayasa Perangkat Lunak

Kode

A

Pertemuan

16

Nama Mahasiswa

Mohammad Shahbana Satriawan

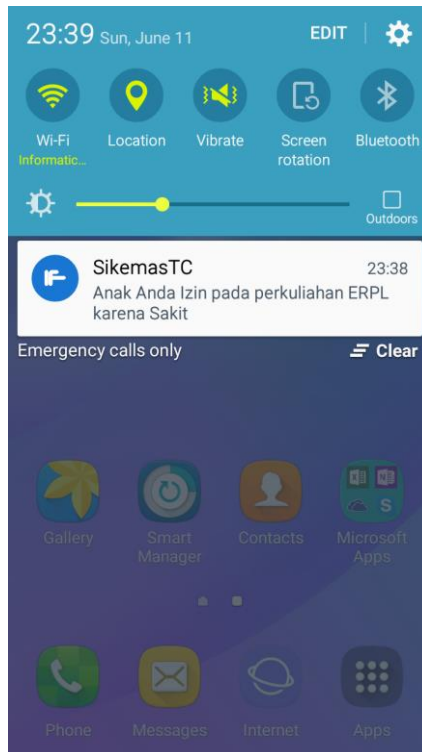
Status Kehadiran

I

Alasan

Sakit

Gambar 5.61 Mengganti Status Kehadiran Mahasiswa



**Gambar 5.62 Notifikasi kepada Orang Tua**

### **5.2.5.2 Pengujian Fungsionalitas Notifikasi Perubahan Jadwal Perkuliahan**

<b>No. Pengujian</b>	<b>UJ-34.UC-35</b>
<b>Nama</b>	Notifikasi Perubahan Jadwal Perkuliahan
<b>Kondisi Awal</b>	Dosen mengganti jadwal perkuliahan
<b>Data Uji</b>	Perkuliahan ERPL A, Pertemuan: 16, Tanggal: 12-06-2017, Jam 7.30, Ruangan: IF-101
<b>Prosedur</b>	Dosen mengganti jadwal perkuliahan dengan data uji

Hasil yang diharapkan	Sistem mengirimkan notifikasi ke perangkat milik mahasiswa
Hasil yang diperoleh	Sistem berhasil mengirimkan notifikasi ke perangkat milik mahasiswa
Hasil Pengujian	Berhasil

Ganti Jadwal

Nama

Ekonomi Reayasa Perangkat Lunak

Kode

A

Pertemuan

16

Pilih tanggal pengganti

12/06/2017

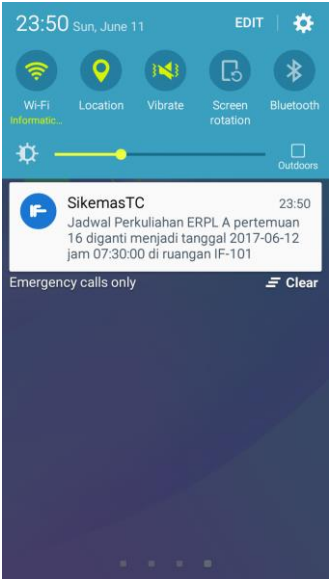
Waktu

07:30:00

Ruangan

IF-101

Gambar 5.63 Pengubahan Jadwal Perkuliahan



Gambar 5.64 Notifikasi Penggantian Jadwal



### 5.3 Pengujian Penerimaan Pengguna

Pengujian penerimaan pengguna ini dilakukan untuk mengetahui tingkat kecocokan aplikasi terhadap pengguna dan keadaan sebenarnya. Kegunaan yang diuji meliputi fitur-fitur yang ada pada tiap aktor.

Pengujian dilakukan dengan memberikan kesempatan pada pengguna untuk mencoba sendiri melakukan aplikasi yang telah dikembangkan dengan arahan dari pengembang aplikasi. Pengujian dilakukan kepada tiga aktor berbeda. . Setelah pengujian selesai, pengguna diminta untuk mengisi kuesioner yang diberikan untuk mengetahui tanggapan pengguna.

Penilaian kuesioner ini memiliki skala 1-5 dimana semakin kecil angka skala yang dipilih berarti semakin tidak setuju sedangkan semakin besar nilai angka skala yang dipilih berarti semakin setuju dengan pernyataan kuesioner.

#### 5.3.1 Pengujian Penerimaan Pengguna Pada Aktor Mahasiswa

Pengujian dilakukan kepada Mahasiswa Teknik Informatika. Kuisiomer, Hasil Kuisiomer dan Rekapitulasi dapat dilihat pada Tabel 5.35, Tabel 5.36 dan Tabel 5.37

**Tabel 5.35 Kuisiomer Aktor Mahasiswa**

No	Pertanyaan	Penilaian				
		1	2	3	4	5
1	Sistem Kehadiran Mahasiswa(SIKEMAS) dapat dioperasikan dengan mudah					
2	SIKEMAS memudahkan mahasiswa melihat					

No	Pertanyaan	Penilaian				
		1	2	3	4	5
	jadwal kelas?					
3	SIKEMAS memudahkan mahasiswa melihat riwayat kehadiran					
4	Menu perkuliahan aktif dan list perkuliahan mudah dipahami					
5	Fitur yang ada di aplikasi ini sesuai dengan kebutuhan mahasiswa tentang informasi jadwal dan kehadiran perkuliahan					

Tabel 5.36 Hasil Kuisioner Aktor Mahasiswa

No	Nama	Nomor Pertanyaan Kuesioner				
		1	2	3	4	5
1	Burhanuddin Rasyid	4	5	4	4	4
2	Mardiana Sekarsari	5	4	3	4	5
3	Didit Sepiyanto	5	4	4	5	5
4	Ibnu Prayogi	4	4	3	4	4
5	Relaci Aprilia	5	4	5	5	5

**Tabel 5.37 Rekapitulasi Kuisisioner Mahasiswa**

Pertanyaan	Hasil Penilaian
Pertanyaan 1	4.6
Pertanyaan 2	4.2
Pertanyaan 3	3.8
Pertanyaan 4	4.4
Pertanyaan 5	4.6
<b>Rata-Rata</b>	<b>4.32</b>

### 5.3.2 Pengujian Penerimaan Pengguna Pada Aktor Dosen

Pengujian dilakukan kepada Dosen Teknik Informatika. Kuisisioner, Hasil Kuisisioner dan Rekapitulasi dapat dilihat pada Tabel 5.38, Tabel 5.39 dan Tabel 5.40

**Tabel 5.38 Kuisisioner Aktor Dosen**

No	Pertanyaan	Penilaian				
		1	2	3	4	5
1	Sistem Kehadiran Mahasiswa(SIKEMAS) dapat dioperasikan dengan mudah					
2	Fitur mengaktifkan kelas dan mengakhiri kelas mudah digunakan					
3	SIKEMAS memudahkan dosen untuk melihat jadwal perkuliahan					

No	Pertanyaan	Penilaian				
		1	2	3	4	5
4	SIKEMAS memudahkan dosen untuk melihat rekap kehadiran mahasiswa					
5	Fitur pengubahan jadwal perkuliahan mudah digunakan					

Tabel 5.39 Hasil Kuisioner Aktor Dosen

No	Nama	Nomor Pertanyaan Kuesioner				
		1	2	3	4	5
1	Wijayanti Nurul K	4	4	4	3	4
2	Dwi Sunaryono	5	4	5	4	5
3	Nurul Fajrin	4	4	4	4	5

Tabel 5.40 Rekapitulasi Kuisioner Dosen

Pertanyaan	Hasil Penilaian
Pertanyaan 1	4.3
Pertanyaan 2	4
Pertanyaan 3	4.3
Pertanyaan 4	3.6
Pertanyaan 5	4.6
<b>Rata-Rata</b>	<b>4.18</b>

### 5.3.3 Pengujian Penerimaan Pengguna Pada Aktor Orang Tua

Pengujian dilakukan kepada Orang Tua dan kakak penulis. Kuisioner, hasil kuisioner dan rekapitulasi dapat dilihat pada Tabel 5.41, Tabel 5.42 dan

Tabel 5.43

**Tabel 5.41 Kuisioner Aktor Orang Tua**

No	Pertanyaan	Penilaian				
		1	2	3	4	5
1	Sistem Kehadiran Mahasiswa(SIKEMAS) dapat dioperasikan dengan mudah					
2	SIKEMAS memudahkan melihat jadwal kuliah anak					
3	SIKEMAS memudahkan monitoring kehadiran anak					
4	Fitur yang ada di aplikasi ini sesuai dengan kebutuhan orang tua tentang informasi jadwal dan kehadiran perkuliahan anak					

**Tabel 5.42 Hasil Kuisioner Aktor Orang Tua**

No	Nama	Nomor Pertanyaan Kuesioner			
		1	2	3	4
1	Lely Fadriati	4	4	4	5
2	Setijono	4	3	4	4
3	Aditya Kurniawan	4	4	5	4

**Tabel 5.43 Rekapitulasi Kuisiener Orang Tua**

Pertanyaan	Hasil Penilaian
Pertanyaan 1	4
Pertanyaan 2	3.6
Pertanyaan 3	4.3
Pertanyaan 4	4.3
<b>Rata-Rata</b>	<b>4.05</b>

#### 5.4 Evaluasi Pengujian

Berdasarkan pengujian yang telah dilakukan, semua pengujian fungsionalitas memberikan hasil yang sesuai dengan skenario yang direncanakan. Evaluasi terhadap pengujian yang dilaksanakan, baik pengujian fungsionalitas dijelaskan sebagai berikut:

1. Fungsionalitas menambah, mengubah, dan menghapus data mahasiswa berjalan sesuai yang diharapkan. Pengujian UJ-01.UC-02, UJ-02.UC-03 dan UJ-02.UC-03 memperlihatkan bahwa sistem memberikan respon yang sesuai dengan masing-masing skenario pengujian.
2. Fungsionalitas menambah, mengubah, dan menghapus data dosen berjalan sesuai yang diharapkan. Pengujian UJ-04-UC.06, UJ-05-UC.07, dan UJ-06-UC.08 memperlihatkan bahwa sistem memberikan respon yang sesuai dengan masing-masing skenario pengujian.
3. Fungsionalitas menambah, mengubah, dan menghapus data orang tua berjalan sesuai yang diharapkan. Pengujian UJ-07.UC-26, UJ-08.UC-27, dan UJ-09.UC-28 memperlihatkan bahwa sistem memberikan respon yang sesuai dengan masing-masing skenario pengujian.
4. Fungsionalitas menambah, mengubah, dan menghapus data ruangan, berjalan sesuai yang diharapkan. Pengujian UJ-10.UC-14, UJ-11.UC-15 dan UJ-11.UC16

memperlihatkan bahwa sistem memberikan respon yang sesuai dengan masing-masing skenario pengujian.

5. Fungsionalitas menambah, mengubah, dan menghapus data matakuliah, berjalan sesuai yang diharapkan. Pengujian UJ-13.UC-10, UJ-14.UC-11, dan UJ-15.UC-12 memperlihatkan bahwa sistem memberikan respon yang sesuai dengan masing-masing skenario pengujian.
6. Fungsionalitas menambah, mengubah, dan menghapus data kelas, berjalan sesuai yang diharapkan. Pengujian UJ-16.UC-18, UJ-17.UC-19, dan UJ-18.UC-20 memperlihatkan bahwa sistem memberikan respon yang sesuai dengan masing-masing skenario pengujian.
7. Fungsionalitas menambah peserta kelas, melihat detail perkuliahan dan melihat rekap kehadiran berjalan sesuai yang diharapkan. Pengujian UJ-19.UC-29 dan UJ-20.UC-21 dan UJ-21.UC-23 memperlihatkan bahwa sistem memberikan respon yang sesuai dengan skenario pengujian.
8. Fungsionalitas yang dimiliki oleh dosen, baik yang merupakan aplikasi web maupun aplikasi perangkat bergerak, semua berjalan lancar. Pengujian UJ-22.UC-36, UJ-23.UC-36, UJ-24.UC-21, UJ-25.UC-23, UJ-26.UC-23, dan UJ-27.UC22, memperlihatkan bahwa sistem memberikan respon yang sesuai dengan skenario pengujian.
9. Fungsionalitas yang dimiliki oleh mahasiswa, semua berjalan lancar. Pengujian UJ-28.UC-21, UJ-29.UC-21, dan UJ-30.UC-23 memperlihatkan bahwa sistem memberikan respon yang sesuai dengan skenario pengujian.
10. Fungsionalitas yang dimiliki oleh orang tua, semua berjalan lancar. Pengujian UJ-31.UC-21, dan UJ-32.UC-33 memperlihatkan bahwa sistem memberikan respon yang sesuai dengan skenario pengujian.

11. Fungsionalitas Notifikasi berjalan lancar. Pengujian UJ-33.UC-35 dan UJ-34.UC-35 memperlihatkan bahwa system memberikan respon sesuai dengna skenario pengujian.
12. Pengujian penerimaan pengguna pada aktor mahasiswa memiliki nilai rata-rata 4.32 dari nilai maksimal 5 yang berarti dapat diterima oleh pengguna.
13. Pengujian penerimaan pengguna pada aktor dosen memiliki nilai rata-rata 4.18 dari nilai maksimal 5 yang berarti dapat diterima oleh pengguna.
14. Pengujian penerimaan pengguna pada aktor orang tua memiliki nilai rata-rata 4.05 dari nilai maksimal 5 yang berarti dapat diterima oleh pengguna.



## **BAB VI**

### **KESIMPULAN DAN SARAN**

Bab ini membahas mengenai kesimpulan yang dapat diambil dari hasil pengujian yang telah dilakukan sebagai jawaban dari rumusan masalah yang dikemukakan. Selain kesimpulan, juga terdapat saran yang ditujukan untuk pengembangan perangkat lunak di masa mendatang.

#### **6.1 Kesimpulan**

1. Aplikasi *Mobile* Sistem Kehadiran Mahasiswa berbasis *Android* berhasil dibuat
2. Aplikasi *Web* dan *Android* Sistem Kehadiran Mahasiswa berhasil diintegrasikan melalui RestApi yang dibangun dengan kerja Laravel sehingga. data-data yang diintegrasikan, diterima, diolah dan dikirimkan bernilai benar.
3. Firebase Cloud Messaging sangat membantu dalam pembuatan dan pengiriman notifikasi karena sangat cepat dan mudah. Dapat dilihat pada pengujian UJ-33.35 dan UJ-34.35

#### **6.2 Saran**

1. Perlunya integrasi sistem yang sudah dibuat dengan sistem akademik.
2. Dapat ditambahkan fitur komunikasi antara dosen dengan orang tua, agar dapat melaporkan masalah mahasiswa secara langsung kepada orang tua.

*[Halaman ini sengaja dikosongkan]*

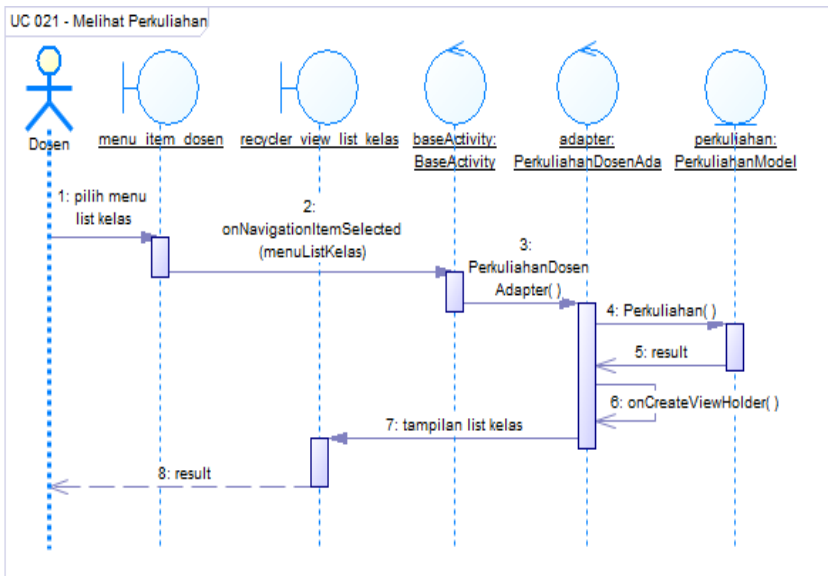
## DAFTAR PUSTAKA

- [1] Muhammad Yusuf, R. V. Hari Ginardi, and A. S. Ahmadiyah, “Rancang Bangun Aplikasi Absensi Perkuliahan Mahasiswa dengan Pengenalan Wajah.”
- [2] N. A. Muhammad, F. Samopa, and R. P. Wibowo, “Pembuatan Aplikasi Presensi Perkuliahan Berbasis Fingerprint (Studi Kasus : Jurusan Sistem Informasi Institut Teknologi Sepuluh Nopember Surabaya),” *J. Tek. ITS*, vol. 2, no. 3, pp. A465–A469, Dec. 2013.
- [3] miftah, “TONGSIS, APLIKASI MONITORING SISWA,” 22-May-2017. [Online]. Available: <https://portal.bandung.go.id/p/bVvn>. [Accessed: 29-May-2017].
- [4] “Profil | Teknik Informatika ITS.” [Online]. Available: <http://if.its.ac.id/sejarah-teknik-informatika/>. [Accessed: 29-May-2017].
- [5] “Sistem,” *Wikipedia bahasa Indonesia, ensiklopedia bebas*. 20-Dec-2016.
- [6] “Arti kata hadir - Kamus Besar Bahasa Indonesia (KBBI) Online.” [Online]. Available: <http://kbbi.web.id/hadir>. [Accessed: 04-Jun-2017].
- [7] “Task Scheduling - Laravel - The PHP Framework For Web Artisans.” [Online]. Available: <https://laravel.com/docs/5.4/scheduling>. [Accessed: 05-Jun-2017].
- [8] “brozot/laravel-fcm - Packagist.” [Online]. Available: <https://packagist.org/packages/brozot/laravel-fcm>. [Accessed: 05-Jun-2017].
- [9] “Transmitting Network Data Using Volley | Android Developers.” [Online]. Available: <https://developer.android.com/training/volley/index.html>. [Accessed: 05-Jun-2017].
- [10] “JSON,” *Wikipedia bahasa Indonesia, ensiklopedia bebas*. 02-Jun-2017.

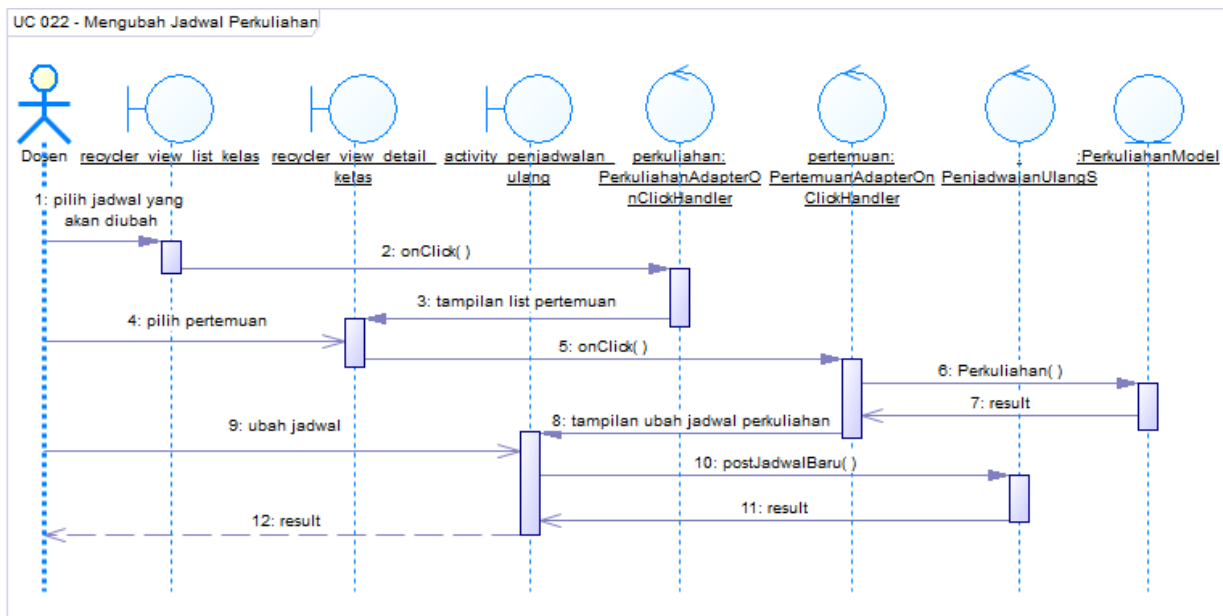
[11]“SQLite,” *Wikipedia*. 04-Jun-2017.

[12]“MySQL,” *Wikipedia bahasa Indonesia, ensiklopedia bebas*. 20-Aug-2016.

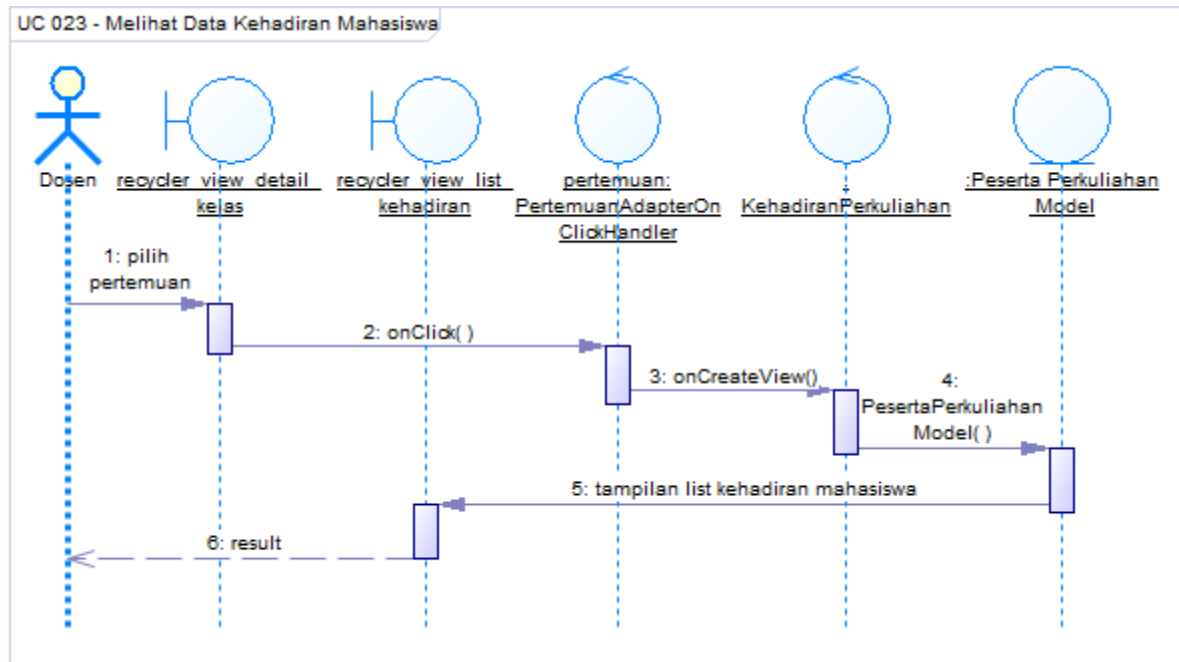
## LAMPIRAN



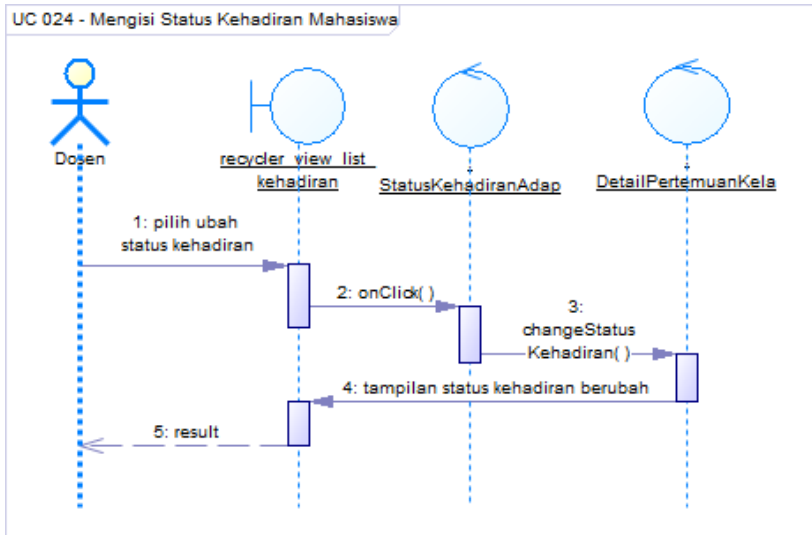
Gambar 0.1 Diagram Sekuens UC-021



Gambar 0.2 Diagram Sekuens UC-022



Gambar 0.3 Diagram Sekuens UC-023



**Gambar 0.4 Diagram Sekuens UC-024**



## BIODATA PENULIS



Mohammad Shahbana Satriawan menempuh pendidikan SMP di SMP Negeri 1 Surabaya, SMA di SMA Negeri 5 Surabaya, dan S1 Teknik Informatika Fakultas Teknologi Informasi di Institut Teknologi Sepuluh Nopember Surabaya (2013-2017).

Selama masa kuliah, penulis aktif dalam organisasi Himpunan Mahasiswa Teknik Computer (HMTC), di antaranya adalah menjadi staf Departemen Kewirausahaan Minat dan Bakat HMTC ITS 2014-2015. panitia Hubungan Masyarakat Schematics HMTC ITS 2014 dan 2015,

Selama berkuliah, selain dari materi perkuliahan, penulis memperoleh pengalaman dalam hal pemrograman dari beberapa proyek dari luar yang dilakukan bersama dengan beberapa teman kuliah. Penulis dapat dihubungi melalui alamat surel [shahbana.satriawan@gmail.com](mailto:shahbana.satriawan@gmail.com)